

PCON、ACON、SCON、ERC2、ERC3

シリアル通信【Modbus 版】

取扱説明書 第7版

お使いになる前に

この度は、当社の製品をお買い上げ頂き、ありがとうございます。

この取扱説明書はシリアル通信(Modbus)の取扱い方法について解説しており、安全にお使い頂く為に必要な情報を記載しています。

お使いになる前に必ずお読み頂き、十分理解した上で安全にお使い頂きますよう、お願い致します。

製品に同梱のCD/DVDには、当社製品の取扱説明書が収録されています。

製品のご使用につきましては、該当する取扱説明書の必要部分をプリントアウトするか、またはパソコンで表示してご利用ください。

お読みになった後も取扱説明書は、取り扱われる方が、必要な時にすぐ読むことができるように保管してください。

【重要】

- この取扱説明書に記載されている以外の運用はできません。記載されている以外の運用をした結果につきましては、一切の責任を負いかねますのでご了承ください。
- この取扱説明書に記載されている事柄は、製品の改良にともない予告なく変更させて頂く場合があります。
- この取扱説明書の内容について、ご不審やお気付きの点などがありましたら、「アイエイアイお客様センターエイト」もしくは最寄りの当社営業所までお問合せください。
- この取扱説明書の全部または一部を無断で使用・複製することはできません。
- 本書中における会社名、商品名は、各社の商標または登録商標です。

目次

安全ガイド	1
取扱上の注意	9
1 概要	11
2 仕様	12
2.1 通信方式	13
3 通信までの準備	14
3.1 上位が RS232C インタフェースの場合	14
(1) システム構成	14
(2) 配線	15
(3) SIO 変換器(垂直仕様:RCB-TU-SIO-A、水平仕様:RCB-TU-SIO-B)	17
3.2 上位が RS485 インタフェースの場合	18
(1) システム構成	18
(2) 配線	19
3.3 PLC および パソコンの通信コネクタ ピンアサイン(参考)	20
3.4 通信を行うまでの各種設定	21
3.5 軸番号の設定	22
3.6 コントローラの通信速度を設定	23
4 通信	25
4.1 メッセージ送信タイミング	25
4.2 タイムアウトとリトライ	26
4.3 RC コントローラの内部アドレス および データ構造	27
4.3.1 Modbus レジスタの構造	27
4.3.2 Modbus レジスタ詳細	28
(1) アラーム詳細コードの内容(アドレス = 0500 _H)(ALA0)	30
(2) アラームアドレスの内容(アドレス = 0501 _H)(ALA0)	30
(3) アラームコードの内容(アドレス = 0503 _H)(ALC0)	31
(4) アラーム発生時刻の内容(アドレス = 0504 _H)(ALTO)	32
(5) デバイス制御レジスタ 1 内容(アドレス = 0D00 _H)(DRG1)	33
(6) デバイス制御レジスタ 2 内容(アドレス = 0D01 _H)(DRG2)	34
(7) ポジション番号指定レジスタ内容(アドレス = 0D03 _H)(POSR)	35
(8) 通算移動回数の内容(アドレス = 8400 _H)(TMLC)	36
(9) 通算走行距離の内容(アドレス = 8402 _H)(ODOM)	37
(10) 現在時刻の内容(アドレス = 841A _H (SCON-CA)、8420 _H (PCON-CA/CFA)(TIMN))	38

(11) FAN 通算駆動時間の内容(アドレス = 842E _H)(ODOM).....	39
(12) デバイスステータスレジスタ1 内容(アドレス = 9005 _H)(DSS1).....	40
(13) デバイスステータスレジスタ2 内容(アドレス = 9006 _H)(DSS2).....	41
(14) 拡張デバイスステータスレジスタ内容(アドレス = 9007 _H)(DSSE).....	42
(15) システムステータスレジスタ内容(アドレス = 9008 _H)(STAT).....	43
(16) 特殊ポートモジュールレジスタ内容(アドレス = 9012 _H)(SIPM).....	44
(17) ゾーンステータスレジスタ内容(アドレス = 9013 _H)(ZONS).....	45
(18) ポジション番号ステータスレジスタ内容(アドレス = 9014 _H)(POSS).....	46
(19) 拡張システムステータスレジスタの内容(アドレス = 9015 _H)(SSSE).....	47
4.3.3 Modbus ステータスの構造.....	48
4.3.4 Modbus ステータス詳細.....	49
5 Modbus RTU.....	53
5.1 メッセージフレーム(クエリ、レスポンス).....	54
5.2 RTU モード クエリ一覧.....	57
5.3 データ、ステータス読み取り(使用ファンクションコード 03).....	61
5.3.1 レジスタの連続複数読み取り.....	61
5.3.2 アラーム詳細内容の読み取り 《ALA0、ALC0、ALTO》.....	65
5.3.3 ポジションデータの読み取り《PCMD、INP、VCMD、ZNMP、ZNLP、ACMD、DCMD、PPOW、LPOW、CTLF》.....	67
5.3.4 通算移動回数の読み取り 《TLMC》.....	70
5.3.5 通算走行距離の読み取り 《ODOM》(1m 単位).....	72
5.3.6 現在時刻の読み取り 《TIMN》.....	74
5.3.7 ファン通算駆動時間の読み取り 《TFAN》.....	78
5.3.8 現在位置の読み取り 《PNOW》(0.01mm 単位).....	80
5.3.9 現在発生アラームコードの読み取り 《ALMC》.....	82
5.3.10 I/O ポート入力信号状態読み取り 《DIPM》.....	84
5.3.11 I/O ポート出力信号状態の読み取り 《DOPM》.....	88
5.3.12 コントローラ状態信号の読み取り 1 《DSS1》.....	92
5.3.13 コントローラ状態信号の読み取り 2 《DSS2》.....	94
5.3.14 コントローラ状態信号の読み取り 3 《DSSE》.....	96
5.3.15 コントローラ状態信号の読み取り 4 《STAT》.....	98
5.3.16 現在速度の読み取り 《VNOW》.....	100
5.3.17 電流値の読み取り 《CNOW》.....	102
5.3.18 偏差の読み取り 《DEVI》.....	104
5.3.19 電源投入後の積算時間の読み取り 《STIM》.....	106
5.3.20 特殊入力ポートの入力信号状態の読み取り 《SIPM》.....	108
5.3.21 ゾーン出力信号の状態読み取り 《ZONS》.....	110

5. 3. 22	位置決め完了ポジションNo.の読み取り 《POSS》	112
5. 3. 23	コントローラ状態信号の読み取り 5 《SSSE》	114
5. 3. 24	カフィードバックデータの読み取り 《FBFC》・・SCON-CA 専用	116
5. 4	動作指令 及び、データ書き換え(使用ファンクションコード 05)	119
5. 4. 1	コイルへの書込み	119
5. 4. 2	セーフティ速度有効/無効切替 《SFTY》	120
5. 4. 3	サーボ ON/OFF 《SON》	122
5. 4. 4	アラームリセット 《ALRS》	124
5. 4. 5	ブレーキ強制解除 《BKRL》	126
5. 4. 6	一時停止 《STP》	128
5. 4. 7	原点復帰 《HOME》	130
5. 4. 8	位置決め動作起動指令 《CSTR》	132
5. 4. 9	ジョグ/インチング切替 《JISL》	134
5. 4. 10	ティーチモード指令 《MOD》	136
5. 4. 11	ポジションデータ取込み指令 《TEAC》	138
5. 4. 12	ジョグ+指令 《JOG+》	140
5. 4. 13	ジョグ-指令 《JOG-》	142
5. 4. 14	スタートポジション 0~7《ST0~ST7》移動指令 (PIO パターン 4, 5 限定)	144
5. 4. 15	ロードセルキャリブレーション指令 《CLBR》・・専用ロードセル接続が必要	146
5. 4. 16	PIO/Modbus 切替設定 《PMSL》	148
5. 4. 17	減速停止 《STOP》	150
5. 5	制御情報の直接書き込み(使用ファンクションコード 06)	153
5. 5. 1	レジスタへの書込み	153
5. 6	位置決めデータの直接書き込み(使用ファンクションコード 10)	157
5. 6. 1	直値移動指令	157
5. 6. 2	ポジションテーブルデータ書込み	175
6	Modbus ASCII	183
6. 1	メッセージフレーム(クエリ、レスポンス)	184
6. 2	ASCIIコード表	187
6. 3	ASCIIモード クエリ一覧	188
6. 4	データ、ステータス読み取り(使用ファンクションコード 03)	193
6. 4. 1	レジスタの連続複数読み取り	193
6. 4. 2	アラーム詳細内容の読み取り 《ALAO、ALCO、ALTO》	197
6. 4. 3	ポジションデータの読み取り《PCMD,INP,VCMD,ZNMP,ZNLP,ACMD,DCMD,PPOW,LPOW,CTLF》	199
6. 4. 4	通算移動回数の読み取り 《TLMC》	202
6. 4. 5	通算走行距離の読み取り 《ODOM》(1m 単位)	204

6.4.6	現在時刻の読み取り《TIMN》	206
6.4.7	ファン通算駆動時間の読み取り《TFAN》	210
6.4.8	現在位置の読み取り(0.01mm 単位)モニタ《PNOW》	212
6.4.9	現在発生アラームコードの読み取り《ALMC》	214
6.4.10	I/Oポート入力信号の状態読み取り《DIPM》	216
6.4.11	I/Oポート出力信号の状態読み取り《DOPM》	220
6.4.12	コントローラ状態信号の読み取り1《DSS1》	224
6.4.13	コントローラ状態信号の読み取り2《DSS2》	226
6.4.14	コントローラ状態信号の読み取り3《DSSE》	228
6.4.15	コントローラ状態信号の読み取り4《STAT》	230
6.4.16	現在速度の読み取り《VNOW》	232
6.4.17	電流値の読み取り《CNOW》	234
6.4.18	偏差の読み取り《DEVI》	236
6.4.19	電源投入後の積算時間の読み取り《STIM》	238
6.4.20	特殊入力ポートの入力信号状態の読み取り《SIPM》	240
6.4.21	ゾーン出力信号の状態の読み取り《ZONS》	242
6.4.22	位置決め完了ポジションNo.照会《POSS》	244
6.4.23	コントローラ状態信号の読み取り5《SSSE》	246
6.4.24	カフィードバックデータの読み取り《FBFC》・・SCON-CA 専用	248
6.5	動作指令および、データ書き換え(使用ファンクションコード 05)	251
6.5.1	コイルへの書込み	251
6.5.2	セーフティ速度有効/無効切替《SFTY》	252
6.5.3	サーボ ON/OFF《SON》	254
6.5.4	アラームリセット《ALRS》	256
6.5.5	ブレーキ強制解除《BKRL》	258
6.5.6	一時停止《STP》	260
6.5.7	原点復帰《HOME》	262
6.5.8	位置決め動作起動指令《CSTR》	264
6.5.9	ジョグ/インテグ切替《JISL》	266
6.5.10	ティーチモード指令《MOD》	268
6.5.11	ポジションデータ取込み指令《TEAC》	270
6.5.12	ジョグ+指令《JOG+》	272
6.5.13	ジョグ-指令《JOG-》	274
6.5.14	スタートポジション0~7《ST0~ST7》(PIOパターン4、5 限定)	276
6.5.15	ロードセルキャリブレーション指令《CLBR》・・専用ロードセル接続が必要	278
6.5.16	PIO/Modbus 切替設定《PMSL》	280

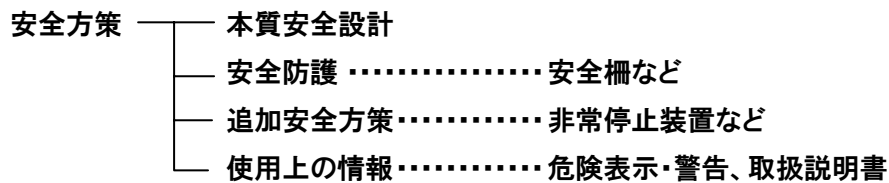
6. 5. 17 減速停止 《STOP》.....	282
6. 6 制御情報の直接書き込み(使用ファンクションコード 06).....	285
6. 6. 1 レジスタへの書き込み	285
6. 7 位置決めデータ直接書き込み(使用ファンクションコード 10).....	289
6. 7. 1 直値移動指令	289
6. 7. 2 ポジションテーブルデータ書き込み.....	307
7 トラブルシューティング	315
7. 1 異常時の返信(例外レスポンス)について.....	316
7. 2 注意事項	319
7. 3 通信がうまくいかない時は	320
8 参考資料.....	327
8. 1 CRC チェック計算	328
8. 2 SIO と PIO の併用システム構成	330

安全ガイド

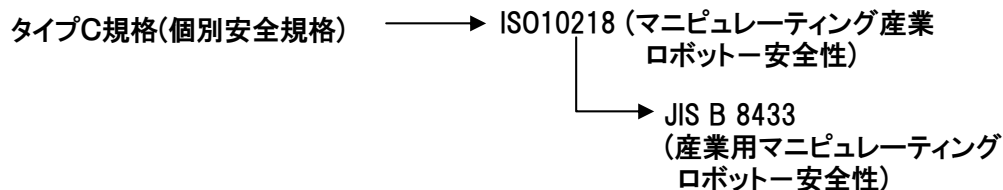
安全ガイドは、製品を正しくお使いいただき、危険や財産の損害を未然に防止するために書かれたものです。製品のお取り扱い前に必ずお読みください。

産業用ロボットに関する法令および規格

機械装置の安全方策としては、国際工業規格 ISO/DIS12100「機械類の安全性」において、一般論として次の4つを規定しています。



これに基づいて国際規格 ISO/IEC で階層別に各種規格が構築されています。産業用ロボットの安全規格は以下のとおりです。



また産業用ロボットの安全に関する国内法は、次のように定められています。

労働安全衛生法 第59条

危険または有害な業務に従事する労働者に対する特別教育の実施が義務付けられています。

労働安全衛生規則

第36条 …… 特別教育を必要とする業務

- | | |
|---|---|
| — | 第31号(教示等) …………… 産業用ロボット(該当除外あり)の教示作業等について |
| — | 第32号(検査等) …………… 産業用ロボット(該当除外あり)の検査、修理、調整作業等について |

第150条 …… 産業用ロボットの使用者の取るべき措置

労働安全衛生規則の産業用ロボットに対する要求事項

作業エリア	作業状態	駆動源の遮断	措置	規定
可動範囲外	自動運転中	しない	運転開始の合図	104 条
			柵、囲いの設置等	150 条の 4
可動範囲内	教示等の作業時	する (運転停止含む)	作業中である旨の表示等	150 条の 3
		しない	作業規定の作成	150 条の 3
			直ちに運転を停止できる措置	150 条の 3
			作業中である旨の表示等	150 条の 3
			特別教育の実施	36 条 31 号
	作業開始前の点検等		151 条	
	検査等の作業時	する	運転を停止して行う	150 条の 5
			作業中である旨の表示等	150 条の 5
		しない (やむをえず運転中に行う場合)	作業規定の作成	150 条の 5
			直ちに運転停止できる措置	150 条の 5
作業中である旨の表示等			150 条の 5	
特別教育の実施 (清掃・給油作業を除く)	36 条 32 号			

当社の産業用ロボット該当機種

労働省告示第 51 号および労働省労働基準局長通達(基発第 340 号)により、以下の内容に該当するものは、産業用ロボットから除外されます。

- (1)単軸ロボットでモータワット数が 80W 以下の製品
- (2)多軸組合せロボットで X・Y・Z 軸が 300mm 以内、かつ回転部が存在する場合はその先端を含めた最大可動範囲が 300mm 立方以内の場合
- (3)多関節ロボットで可動半径および Z 軸が 300mm 以内の製品

当社カタログ掲載製品のうち産業用ロボットの該当機種は以下のとおりです。

1. 単軸ロボシリンダ
RCS2/RCS2CR-SS8口/RCS3 でストローク 300mm を超えるもの
2. 単軸ロボット
次の機種でストローク 300mm を超え、かつモータ容量 80W を超えるもの ISA/ISPA, ISDA/
ISA/ISB/ISPA/ISPB, SSPA, ISDA/ISDB/ISPA/ISPDB, SSPDA, ISWA/ISPWA, IF, FS, NS
3. リニアサーボアクチュエータ
ストローク 300mm を超える全機種
4. 直交ロボット
1~3 項の機種のいずれかを 1 軸でも使用するもの および CT4
5. IX スカラロボット
アーム長 300mm を超える全機種
(IX-NNN1205/1505/1805/2515、NNW2515、NNC1205/1505/1805/2515 を除く全機種)

当社製品の安全に関する注意事項

ロボットのご使用にあたり、各作業内容における共通注意事項を示します。製品ごとの取扱説明書とあわせてお読みください。

No	作業内容	注意事項
1	機種選定	<ul style="list-style-type: none"> ●本製品は、高度な安全性を必要とする用途には企画、設計されていませんので、人命を保証できません。従って、次のような用途には使用しないでください。 <ul style="list-style-type: none"> ①人命および身体の維持、管理などに関わる医療機器 ②人の移動や搬送を目的とする機構、機械装置(車両・鉄道施設・航空施設など) ③機械装置の重要保安部品(安全装置など) ●製品は仕様範囲外で使用しないでください。著しい寿命低下を招き、製品故障や設備停止の原因となります。 ●次のような環境では使用しないでください。 <ul style="list-style-type: none"> ①可燃性ガス、発火物、引火物、爆発物などが存在する場所。 ②放射能に被爆する恐れがある場所 ③周囲温度や相対湿度が仕様の範囲を超える場所 ④直射日光や大きな熱源からの輻射熱が加わる場所 ⑤温度変化が急激で結露するような場所 ⑥腐食性ガス(硫酸、塩酸など)がある場所 ⑦塵埃、塩分、鉄粉が多い場所 ⑧本体に直接振動や衝撃が伝わる場所 ●垂直に使用するアクチュエータは、ブレーキ付きの機種を選定してください。ブレーキがない機種を選定すると、電源をオフしたとき可動部が落下し、けがやワークの破損などの事故を起こすことがあります。
2	運搬	<ul style="list-style-type: none"> ●重量物を運ぶ場合には 2 人以上で運ぶ、または、クレーンなどを使用してください。 ●2 人以上で作業を行う場合は、主と従の関係を明確にし、声を掛け合い、安全を確認しながら作業を行ってください。 ●運搬時は、持つ位置、重量、重量バランスを考慮し、ぶついたり落下しないように十分な配慮をしてください。 ●運搬は適切な運搬手段を用いて行ってください。 クレーンの使用可能なアクチュエータには、アイボルトが取り付けられているか、または取付用タップ穴が用意されていますので、個々の取扱説明書に従って行ってください。 ●梱包の上には乗らないでください。 ●梱包が変形するような重い物は載せないでください。 ●能力が 1t 以上のクレーンを使用する場合は、クレーン操作、玉掛けの有資格者が作業を行ってください。 ●クレーンなどを使用する場合は、クレーンなどの定格荷重を超える荷物は絶対に吊らないでください。 ●荷物にふさわしい吊具を使用してください。吊具の切断荷重などに安全を見込んでください。また、吊具に損傷がないか確認してください。 ●吊った荷物に人は乗らないでください。 ●荷物を吊ったまま放置しないでください。 ●吊った荷物の下に入らないでください。
3	保管・保存	<ul style="list-style-type: none"> ●保管・保存環境は設置環境に準じますが、特に結露の発生がないように配慮してください。 ●地震などの天災により、製品の転倒、落下がおきないように考慮して保管してください。





No	作業内容	注意事項
4	据付け ・立ち上げ	<p>(1) ロボット本体・コントローラ等の設置</p> <ul style="list-style-type: none"> ●製品(ワークを含む)は、必ず確実な保持、固定を行ってください。製品の転倒、落下、異常動作等によって破損およびけがををする恐れがあります。また、地震などの天災による転倒や落下にも備えてください。 ●製品の上に乗ったり、物を置いたりしないでください。転倒事故、物の落下によるけがや製品破損、製品の機能喪失・性能低下・寿命低下などの原因となります。 ●次のような場所で使用する場合は、しや蔽対策を十分行ってください。 <ul style="list-style-type: none"> ①電気的なノイズが発生する場所 ②強い電界や磁界が生じる場所 ③電源線や動力線が近傍を通る場所 ④水、油、薬品の飛沫がかかる場所 <p>(2) ケーブル配線</p> <ul style="list-style-type: none"> ●アクチュエータ～コントローラ間のケーブルやティーチングツールなどのケーブルは当社の純正部品を使用してください。 ●ケーブルに傷をつけたり、無理に曲げたり、引っ張ったり、巻きつけたり、挟み込んだり、重いものを載せたりしないでください。漏電や導通不良による火災、感電、異常動作の原因になります。 ●製品の配線は、電源をオフして誤配線がないように行ってください。 ●直流電源(+24V)を配線する時は、+／-の極性に注意してください。接続を誤ると火災、製品故障、異常動作の恐れがあります。 ●ケーブルコネクタの接続は、抜け・ゆるみのないように確実に行ってください。火災、感電、製品の異常動作の原因になります。 ●製品のケーブルの長さを延長または短縮するために、ケーブルの切断再接続は行わないでください。火災、製品の異常動作の原因になります。 <p>(3) 接地</p> <ul style="list-style-type: none"> ●接地は、感電防止、静電気帯電の防止、耐ノイズ性能の向上および不要な電磁放射の抑制には必ず行われなければなりません。 ●コントローラの AC 電源ケーブルのアース端子および制御盤のアースプレートは、必ず線径 0.5mm²(AWG20 相当)以上のより線で接地工事をしてください。保安接地は、負荷に応じた線径が必要です。規格(電気設備技術基準)は D 種(旧第三種、接地抵抗 100Ω 以下)接地工事を施工してください。 <p>(4) 安全対策</p> <ul style="list-style-type: none"> ●2人以上で作業を行う場合は、主と従の関係を明確にし、声を掛け合い、安全を確認しながら作業を行ってください。 ●製品の動作中または動作できる状態の時は、ロボットの可動範囲に立ち入ることができないような安全対策(安全防護柵など)を施してください。動作中のロボットに接触すると死亡または重傷を負うことがあります。 ●運転中の非常事態に対し、直ちに停止することができるように非常停止回路を必ず設けてください。 ●電源投入だけで起動しないよう安全対策を施してください。製品が急に起動し、けがや製品破損の原因になる恐れがあります。 ●非常停止解除や停電後の復旧だけで起動しないよう、安全対策を施してください。人身事故、装置の破損などの原因となります。 ●据付・調整などの作業を行う場合は、「作業中、電源投入禁止」などの表示をしてください。不意の電源投入により感電やけがの恐れがあります。 ●停電時や非常停止時にワークなどが落下しないような対策を施してください。 ●必要に応じて保護手袋、保護めがね、安全靴を着用して安全を確保してください。 ●製品の開口部に指や物を入れしないでください。けが、感電、製品破損、火災などの原因になります。 ●垂直に接地しているアクチュエータのブレーキを解除する時は、自重で落下して手を挟んだり、ワークなどを損傷しないようにしてください。

No	作業内容	注意事項
5	教示	<ul style="list-style-type: none"> ●2人以上で作業を行う場合は、主と従の関係を明確にし、声を掛け合い、安全を確認しながら作業を行ってください。 ●教示作業はできる限り安全防護柵*外から行ってください。やむをえず安全防護柵内で作業する時は、「作業規定」を作成して作業員への徹底を図ってください。 ●安全防護柵内で作業する時は、作業員は手元非常停止スイッチを携帯し、異常発生時にはいつでも動作停止できるようにしてください。 ●安全防護柵内で作業する時は、作業員以外に監視員を置いて、異常発生時にはいつでも動作停止できるようにしてください。また第三者が不用意にスイッチ類を操作することのないよう監視してください。 ●見やすい位置に「作業中」である旨の表示をしてください。 ●垂直に設置しているアクチュエータのブレーキを解除する時は、自重で落下して手を挟んだり、ワークなどを損傷しないようにしてください。 <p>※安全防護柵・・・安全防護柵がない場合は、可動範囲を示します。</p>
6	確認運転	<ul style="list-style-type: none"> ●2人以上で作業を行う場合は、主と従の関係を明確にし、声を掛け合い、安全を確認しながら作業を行ってください。 ●教示およびプログラミング後は、1ステップずつ確認運転をしてから自動運転に移ってください。 ●安全防護柵内で確認運転をする時は、教示作業と同様にあらかじめ決められた作業手順で作業を行ってください。 ●プログラム動作確認は、必ずセーフティ速度で行ってください。プログラムミスなどによる予期せぬ動作で事故をまねく恐れがあります。 ●通電中に端子台や各種設定スイッチに触れないでください。感電や異常動作の恐れがあります。
7	自動運転	<ul style="list-style-type: none"> ●自動運転を開始する前、あるいは停止後の再起動の際には、安全防護柵内に人がいないことを確認してください。 ●自動運転を開始する前には、関連周辺機器がすべて自動運転に入ることのできる状態にあり、異常表示がないことを確認してください。 ●自動運転の開始操作は、必ず安全防護柵外から行うようにしてください。 ●製品に異常な発熱、発煙、異臭、異音が生じた場合は、直ちに停止して電源スイッチをオフしてください。火災や製品破損の恐れがあります。 ●停電した時は電源スイッチをオフしてください。停電復旧時に製品が突然動作し、けがや製品破損の原因になることがあります。

No	作業内容	注意事項
8	保守・点検	<p>2人以上で作業を行う場合は、主と従の関係を明確にし、声を掛け合い、安全を確認しながら作業を行ってください。</p> <ul style="list-style-type: none"> ●作業はできる限り安全防護柵*外から行ってください。やむをえず安全防護柵内で作業する時は、「作業規定」を作成して作業者への徹底を図ってください。 ●安全防護柵内で作業を行う場合は、原則として電源スイッチをオフしてください。 ●安全防護柵内で作業する時は、作業者は手元非常停止スイッチを携帯し、異常発生時にはいつでも動作停止できるようにしてください。 ●安全防護柵内で作業する時は、作業者以外に監視人をおいて、異常発生時にはいつでも動作停止できるようにしてください。また第三者が不用意にスイッチ類を操作することのないよう監視してください。 ●見やすい位置に「作業中」である旨の表示をしてください。 ●ガイド用およびボールネジ用グリースは、各機種の取扱説明書により適切なグリースを使用してください。 ●絶縁耐圧試験は行わないでください。製品の破損の原因になることがあります。 ●垂直に設置しているアクチュエータのブレーキを解除する時は、自重で落下して手を挟んだり、ワークなどを損傷しないようにしてください。 ●サーボオフするとスライダやロッドが停止位置からずれることがあります。不要動作による、けがや損傷をしないようにしてください。 ●カバーや取り外したネジ等は、紛失しないよう注意し、保守・点検完了後は必ず元の状態に戻して使用してください。 不完全な取り付けは、製品破損やけがの原因となります。 <p>※安全防護柵・・・安全防護柵がない場合は、可動範囲を示します。</p>
9	改造	<ul style="list-style-type: none"> ●お客様の独自の判断に基づく改造、分解組立て、指定外の保守部品の使用は行わないでください。
10	廃棄	<ul style="list-style-type: none"> ●製品が使用不能、または不要になって廃棄する場合は、産業廃棄物として適切な廃棄処理をしてください。 ●廃棄のためアクチュエータを取り外す場合は、落下等に考慮し、ネジの取り外しを行ってください。 ●製品の廃棄時は、火中に投じないでください。製品が破裂したり、有毒ガスが発生する恐れがあります。
11	その他	<ul style="list-style-type: none"> ●ペースメーカーなどの医療機器を装着された方は、影響を受ける場合がありますので、本製品および配線には近づかないようにしてください。 ●海外規格への対応は、海外規格マニュアルを確認してください。 ●アクチュエータおよびコントローラは、それぞれの専用取扱説明書に従い、安全に取り扱ってください。

注意表示について

各機種の取扱説明書には、安全事項を以下のように「危険」「警告」「注意」「お願い」にランク分けして表示しています。

レベル	危害・損害の程度	シンボル
危険	取扱いを誤ると、死亡または重傷に至る危険が差し迫って生じると想定される場合	 危険
警告	取扱いを誤ると、死亡または重傷に至る可能性が想定される場合	 警告
注意	取扱いを誤ると、傷害または物的損害の可能性が想定される場合	 注意
お願い	傷害の可能性はないが、本製品を適切に使用するために守っていただきたい内容	 お願い

取扱上の注意

本書は、シリアル通信の手順に限った説明書となっておりますので、他の制御、設置・接続等に関しましてはロボット用コントローラ(以下 RC コントローラ)に付属している取扱説明書をご覧ください。

⚠ 注意

- (1) 製品の使用条件、使用環境、仕様範囲を守ってお使いください。
守られない場合、性能低下や製品の故障を招きます。
- (2) 本仕様書に掲載されていないアドレスまたはファンクションを RC コントローラに送信した場合には、RC コントローラが正常動作できなくなる、または意図しない動きをする可能性があります。指定外のファンクション及びアドレスへの送信は行わないでください。
- (3) RC コントローラは、SIO ポートより 150[msec]以上のブレイク(スペース)信号を検出するとポートレートが 9600[bps]に切り替わる仕様です。
一部のパソコンは通信ポート非オープン時、送信ラインがブレイク(スペース)状態となっているものがあります。このようなパソコンを上位機器として使用する場合、意図せずに RC コントローラのポートレートが 9600[bps]となっている場合がありますのでご注意ください。
- (4) 通信速度等のパラメータ設定は、当社のパソコン対応ソフトなどの専用チューニングツールを用いて行ってください。
- (5) 次の場所で使用する際は、しゃ蔽対策を十分行ってください。措置しない場合は、誤作動を起こす可能性があります。
 - ① 大電流や高磁界が発生している場所
 - ② 溶接作業などアーク放電の生じる場所
 - ③ 静電気などによるノイズが発生する場所
 - ④ 放射能に被曝する可能性がある場所
- (6) 配線を行ったり、各コネクタの抜き差しの際には、上位側、各 RC コントローラの電源を OFF してください。電源を ON したまま行くと感電や部品の破損を招く恐れがあります。

- (7) ノイズによる誤動作を防止する為、通信の配線は動力線や他の制御用配線と分離して配線してください。
- (8) ノイズによる誤動作を防止する為、同一電源路 あるいは同一装置内の電気機器にはノイズ防止対策を施してください。
- (9) Modbus アドレスの 0503_H、9002_Hに出力されるアラームコードは、メッセージレベルのアラームコードも含まれます。当社のコントローラによってはメッセージレベルアラームが発生しないタイプもありますので、メッセージレベルアラームが発生しないコントローラから、発生するコントローラに置き換える場合、アラームレベルにより動作を変える必要があるシステムでは、メッセージレベルアラーム発生時の動作を追加してください。(例 PCON-C から PCON-CA に置き換え)
発生するアラームレベルは、各コントローラ取扱説明書のトラブルシューティングを参照してください。

1 概要

ロボット用コントローラ(以下 RC コントローラ)はホスト(上位コントローラ)とのインタフェースに EIA RS485 に準拠した調歩同期式シリアルバスインタフェースを装備しています。このインタフェースによって、最大 16 軸までのスレーブ(RC コントローラ)を接続^(注 1)し、制御できる SIO リンクシステムを構築することができます。

各軸毎に指令を出すことはもちろん、全てのスレーブに同時に同じ指令を出すことも可能です。

通信プロトコルは Modbus Protocol を採用し、ホストから指令したり、内部情報の参照が行えます。

Modbus Protocol は、仕様が全世界に公開されていますので、ソフトウェア開発が手軽に行えます。

注 1 同一回線上に旧 RC シリーズ(プロトコル T)や RC シリーズ以外の機器を接続することはできません。

シリアル伝送モードには ASCII モード(1 バイト(8 ビット)データを ASCII コード(2 文字)に変換して伝送)と、RTU モード(1 バイト(8 ビット)データをそのまま伝送)の 2 種類がありますが、RC コントローラにおいては 1 パケットごとに伝送モードを判定しており、どちらのモードでも受信可能^(注 2)となっています。ROBONET_RS485 は SIO スルーモードに設定してください。[別冊 ROBONET 取扱説明書参照]

注 2 1 つのネットワーク上では、全てのデバイスを同一モードで使用ください。混在はできません。

☆ 制御可能なコントローラ

- ・ ERC2(SE) / ERC3*
- ・ PCON-C / CA* / CFA* / CG / CF / CY / SE / PL / PO
- ・ ACON-C / CG / CY / SE / PL / PO
- ・ SCON-C / CA
- ・ ROBONET_RS485(RTU モードかつ SIO スルーモード時)

* V0002以降

1.1 DVD に収録されている本製品関連の取扱説明書

番号	名称	管理番号
1	ERC2(PIO) コントローラ一体型アクチュエータ取扱説明書	MJ0158
2	ERC2(SE:SIO) コントローラ一体型アクチュエータ取扱説明書	MJ0159
3	PCON-C/CG/CF コントローラ取扱説明書	MJ0170
4	PCON-CY コントローラ取扱説明書	MJ0156
5	PCON-SE コントローラ取扱説明書	MJ0163
6	PCON-PL/PO コントローラ取扱説明書	MJ0164
7	ACON-C/CG コントローラ取扱説明書	MJ0176
8	ACON-CY コントローラ取扱説明書	MJ0167
9	ACON-SE コントローラ取扱説明書	MJ0171
10	ACON-PL/PO コントローラ取扱説明書	MJ0166
11	SCON-C コントローラ取扱説明書	MJ0161
12	SCON-CA コントローラ取扱説明書	MJ0243
13	ROBONET 取扱説明書	MJ0208
14	ERC3 コントローラ一体型アクチュエータ取扱説明書	MJ0279
15	PCON-CA/CFA コントローラ取扱説明書	MJ0289

2 仕様

項目	方式・条件
インタフェイス	EIA RS485 準拠
通信方式	半二重通信
最大総延長距離	100m
同期方式	調歩同期式
接続形態	1:N 不平衡バス接続 (1 ≤ N ≤ 16)
伝送モード	RTU/ASCII(自動判別) ^(注)
通信速度(bps)	パラメータ設定に次の速度から選択可能 9600,14400,19200,28800,38400 57600,76800,115200,230400
ビット長	8ビット
ストップビット	1ビット
パリティ	なし

注 ROBOTNET は ASCII モードには対応していません。

2.1 通信方式

Modbus Protocol の通信方式は、シングルマスタ/マルチスレーブ方式です。マスタ(上位:以下の例では PLC)が指定したスレーブ(以下の例では C 軸に接続された RC コントローラ)にクエリを発行し、指定されたスレーブは、このクエリを受けて、指定された機能を実行し、レスポンスメッセージを返します。(これで 1 回の通信サイクルが終了します。)

クエリの送信フォーマットは、スレーブのアドレス、要求内容を定義するファンクションコード、データおよびエラーチェックから構成されています。

また、レスポンスメッセージの送信フォーマットは、要求内容の確認ファンクションコード、データおよびエラーチェックから構成されています。クエリとレスポンスメッセージの送信構造を下図に示します。

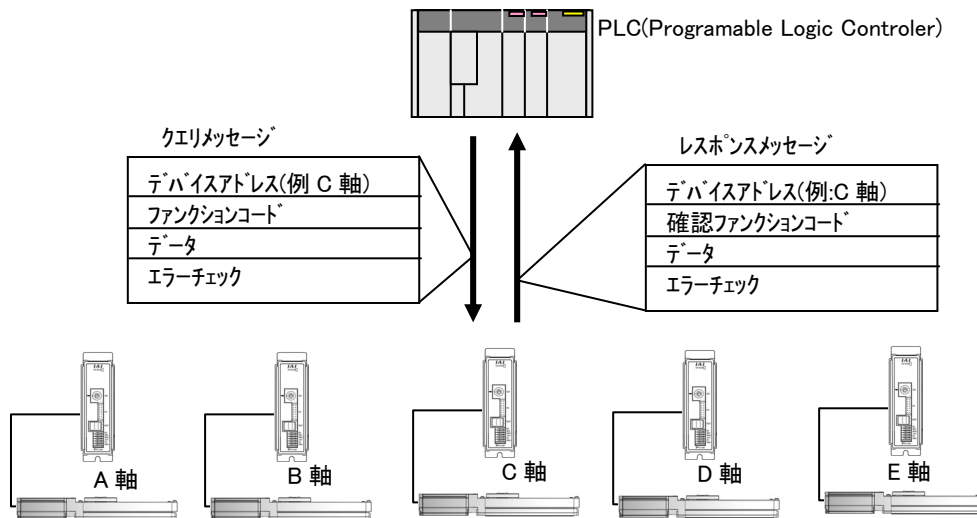
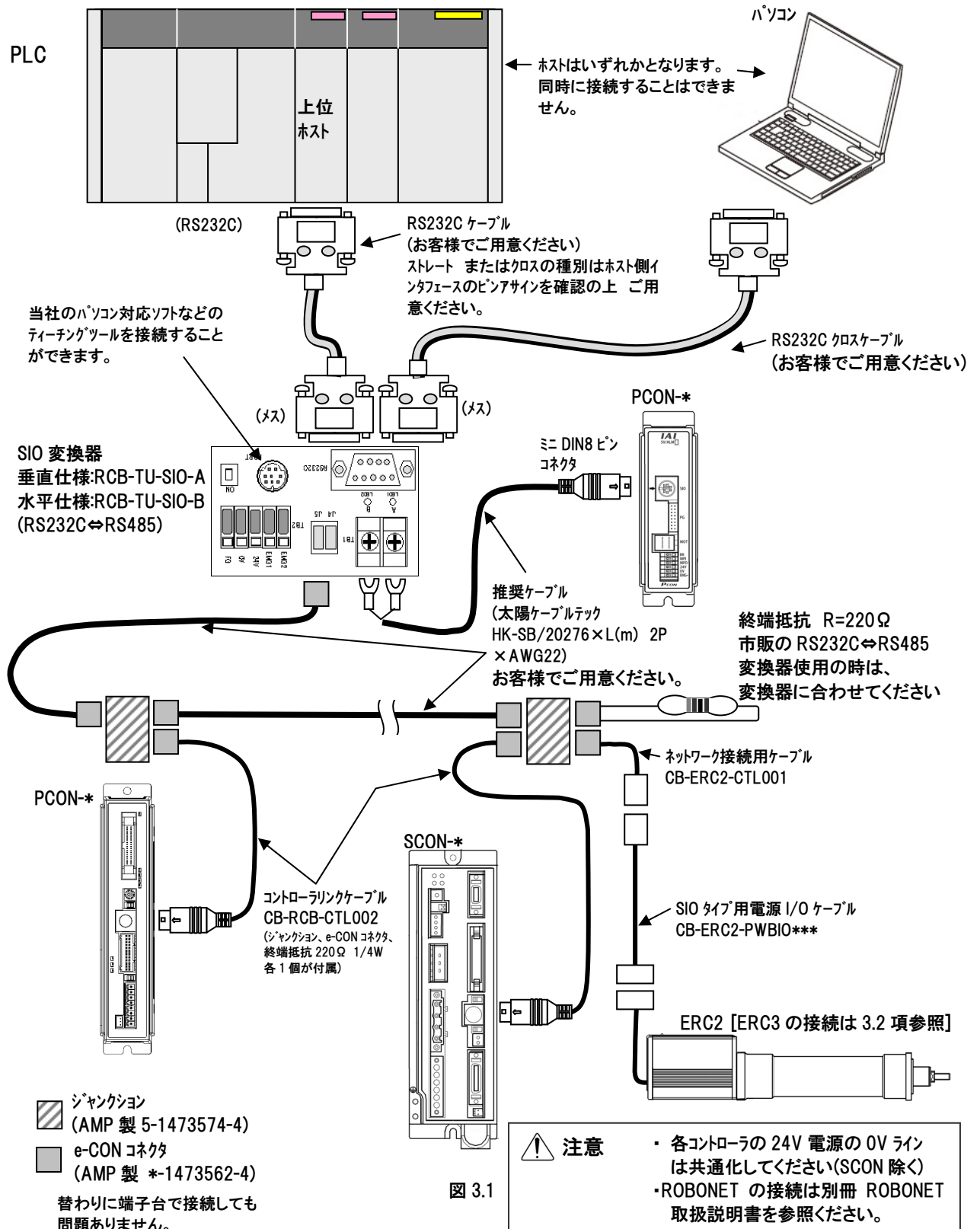


図 2.1

3 通信までの準備

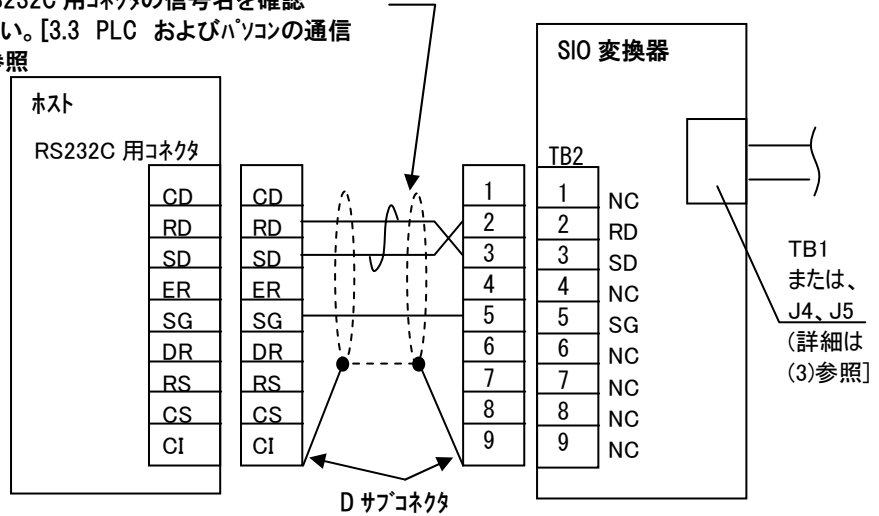
3.1 上位が RS232C インタフェースの場合

(1) システム構成



(2) 配線

RS232C クロスケーブル(市販品等)
必ず、ホスト側の RS232C 用コネクタの信号名を確認して接続してください。[3.3 PLC およびパソコンの通信コネクタ ピンサイン]参照



注意 各コントローラの 24V 電源の 0V ラインは共通化してください(SCON 除く)
ROBONET の接続は、別冊 ROBONET 取扱説明書を参照ください。

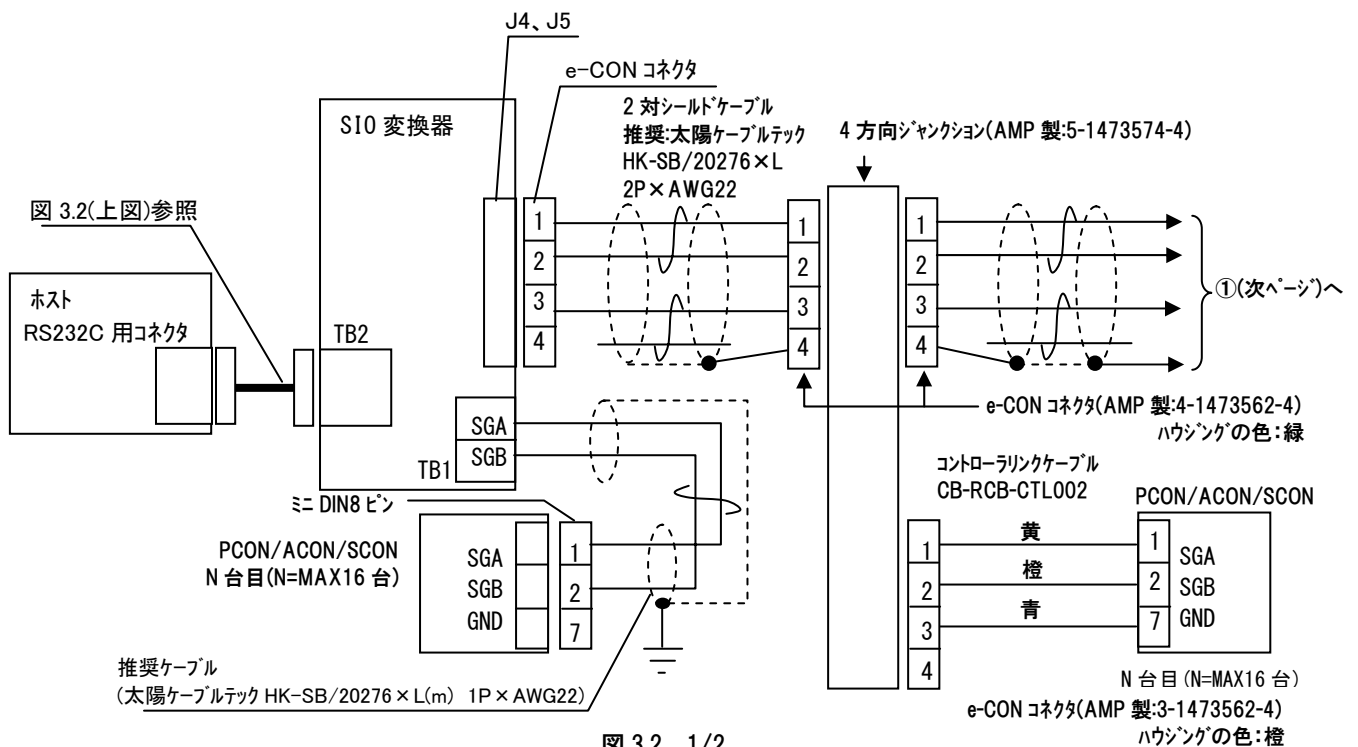
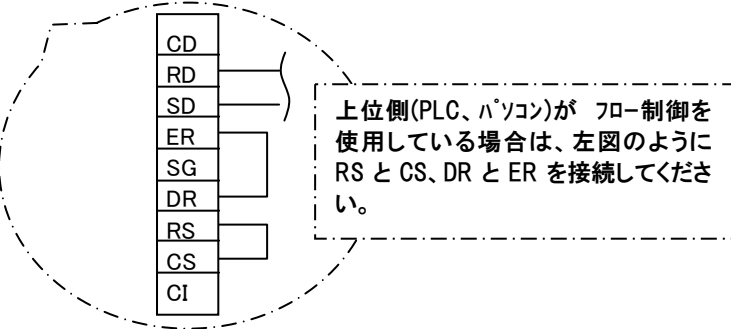


図 3.2_1/2

準備

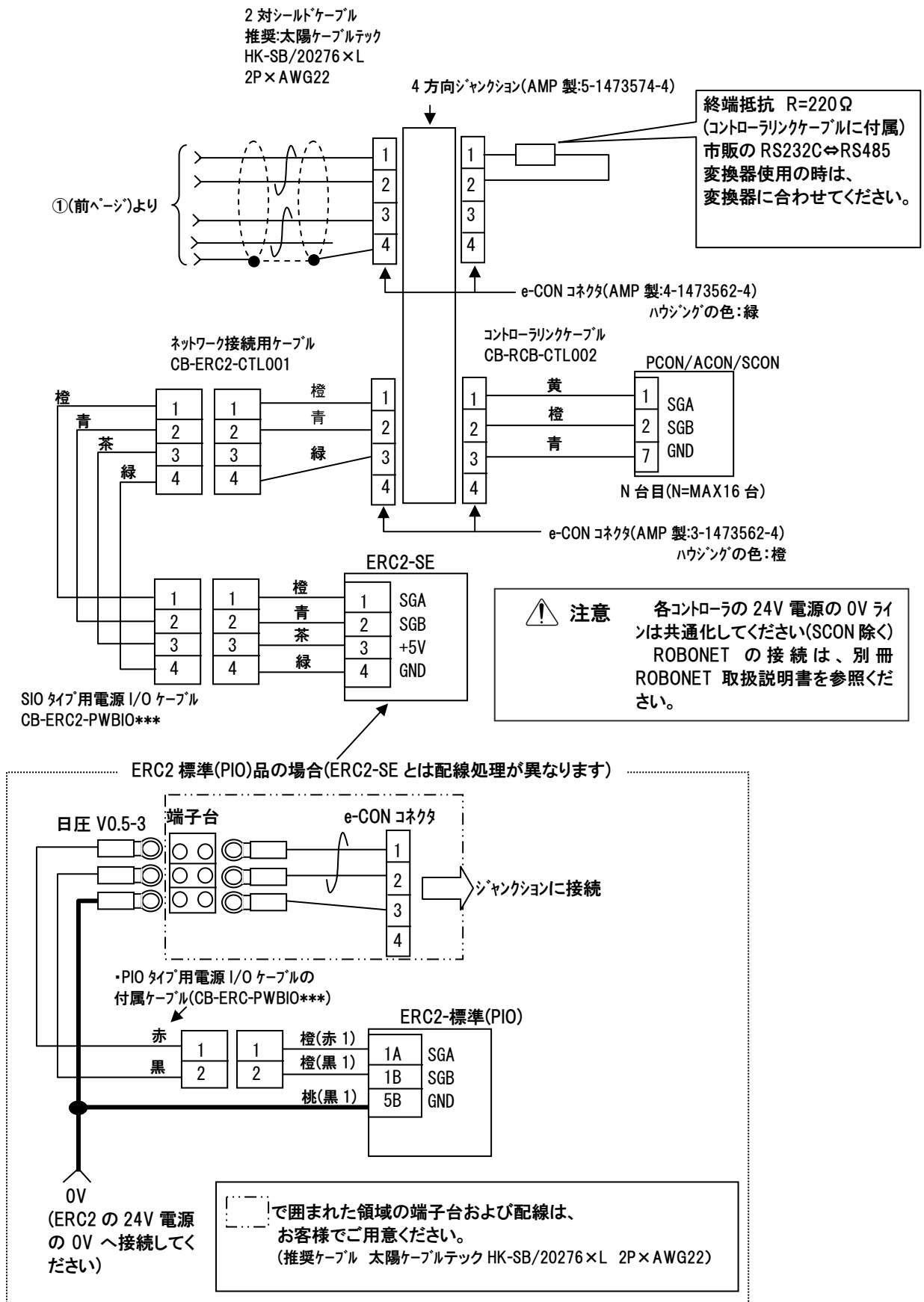
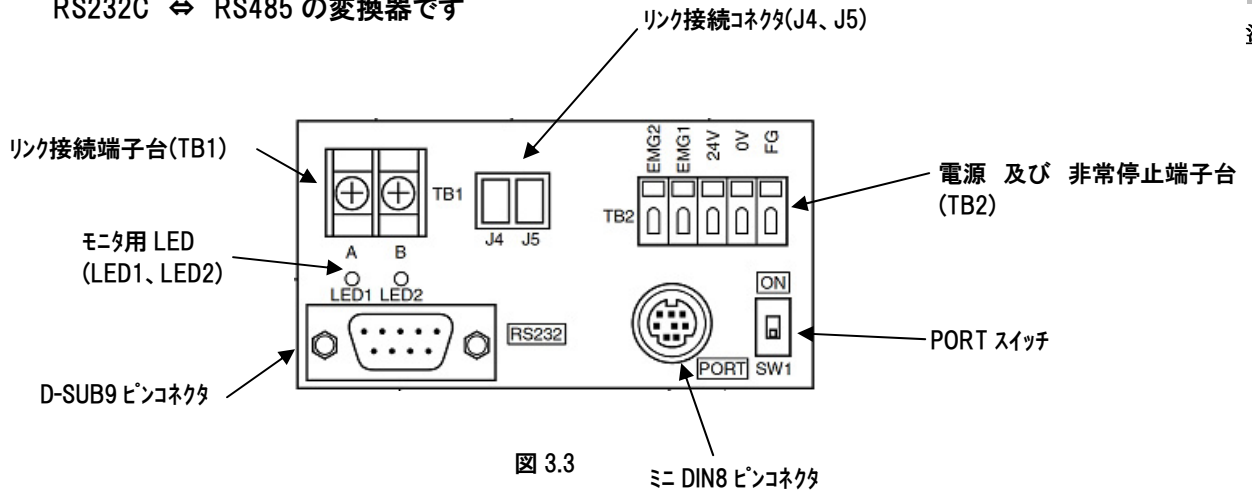


図 3.2_2/2

(3) SIO 変換器(垂直仕様:RCB-TU-SIO-A、水平仕様:RCB-TU-SIO-B)
RS232C ⇔ RS485 の変換器です



◎電源 及び 非常停止端子台(TB2)

- ・EMG1、EMG2: ティーチングホックスの非常停止スイッチの接点出力
PORT スイッチが ON 側でティーチングホックスの非常停止スイッチに接続され、
OFF 側では EMG1 と EMG2 は 短絡されます。
 - ・24V: +24V 電源を供給してください。(消費電流 0.1A 以下)
 - ・0V : 0V 電源を供給してください。(DC24V 仕様のコントローラの 0V と共通にしてください)
 - ・FG : FG を接続する端子です。
- ※適合電線 単線 Φ0.8~1.2mm、
撚り線 AWG18~20(ストリップ長 10mm)

◎リンク接続端子台(TB1)

- RC コントローラとリンク接続するための接続口です。
- ・A: RC コントローラの通信コネクタの 1 ピン(SGA)に接続します。
 - ・B: RC コントローラの通信コネクタの 2 ピン(SGB)に接続します。



◎D-sub9 ピンコネクタ

上位(ホスト)側との接続口です。

◎ミニ DIN8 ピンコネクタ

パソコン対応ソフトなどのティーチングツールとの接続口です。

◎PORT スイッチ

- ・ON: ティーチングツール使用
- ・OFF: ティーチングツール未使用

◎モータ用 LED(LED1、LED2)

- ・LED1: RC コントローラが送信中のときに点灯/点滅します。
- ・LED2: 上位(ホスト)側が送信中のときに点灯/点滅します。

◎リンク接続コネクタ(J4、J5)

RC コントローラとリンク接続するための接続口です。
オプションのリンクケーブル(CB-RCB-CTL002)をそのまま接続できます。

3.2 上位が RS485 インタフェースの場合

(1) システム構成

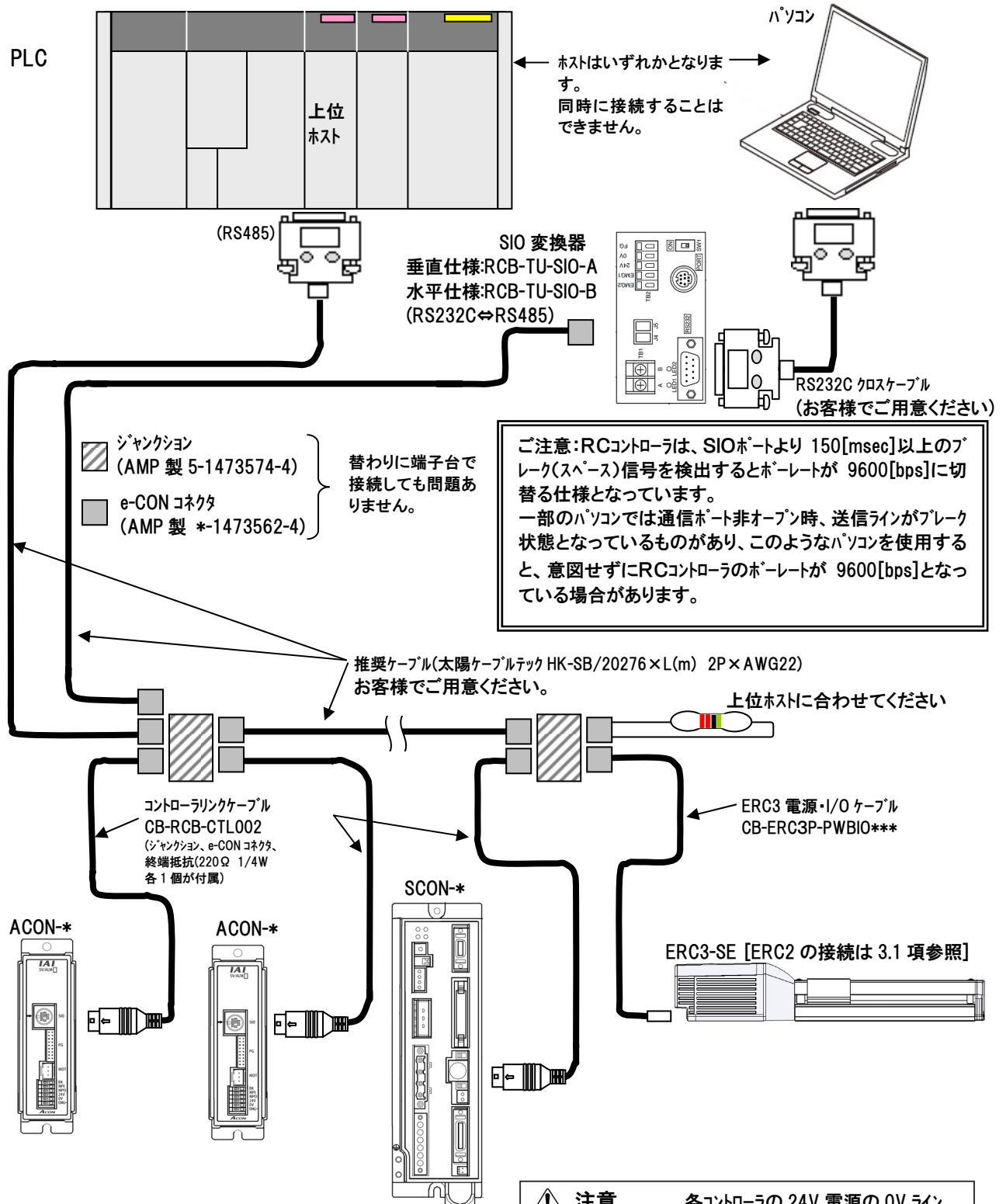


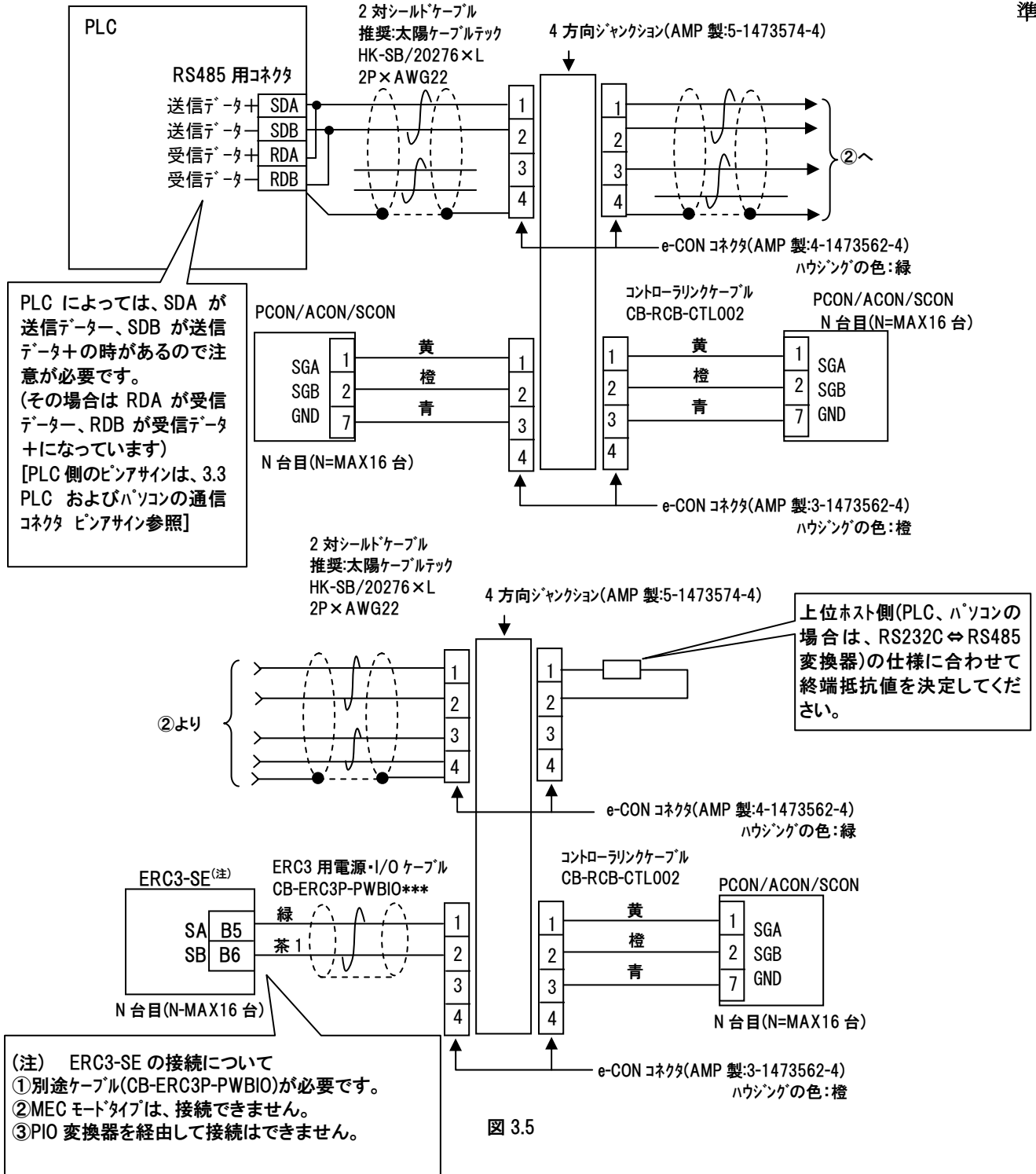
図 3.4

(2) 配線



注意

各コントローラの 24V 電源の 0V ラインは共通化してください(SCON 除く)
ROBONET の接続は、別冊 ROBONET 取扱説明書を参照ください。

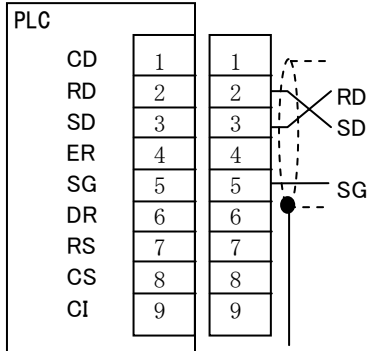


3.3 PLC および パソコンの通信コネクタ ピンサイン(参考)

三菱製 PLC:QJ71C24

RS232C の時

D サブ 9 ピンコネクタ(オス:ケーブル側)

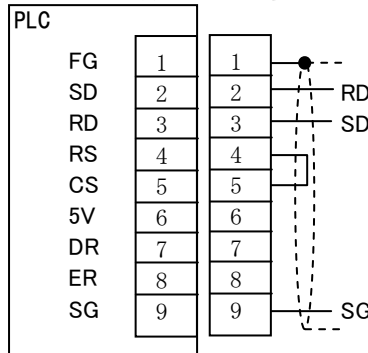


シールドは片側を
コネクタハウジングに接続
または接地

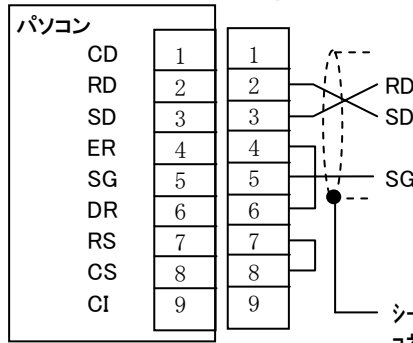
オムロン製 PLC:CJ1W-SCB, SCU

RS232C の時

D サブ 9 ピンコネクタ(オス:ケーブル側)



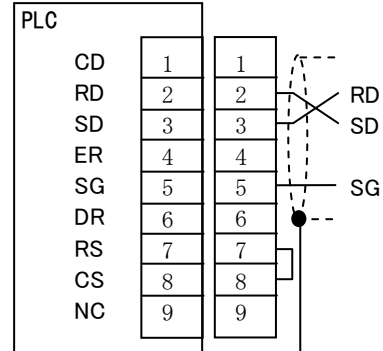
パソコン :RS232C
D サブ 9 ピンコネクタ(メス:ケーブル側)



キーエンス製 PLC:KV-L20R

RS232C の時

D サブ 9 ピンコネクタ(メス:ケーブル側)



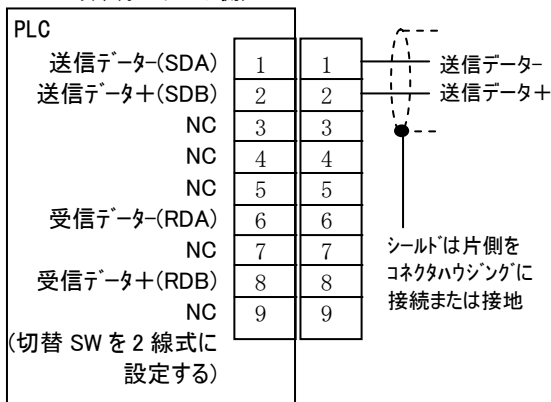
シールドは片側を
コネクタハウジングに
接続または接地

フロー制御を使用する場合は、
RS と CS、DR と ER を接続して
ください。

オムロン製 PLC:CJ1W-SCB, SCU

RS485 の時

D サブ 9 ピンコネクタ(オス:ケーブル側)

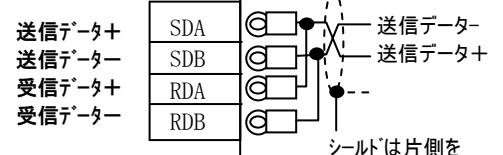


(切替 SW を 2 線式に
設定する)

三菱製 PLC:QJ71C24 RS485 の時

端子台

通信ユニットパネルに信号名が印字されてい
るのでそれに従って配線



シールドは片側を
コネクタハウジングに
接続または接地

キーエンス製 PLC:KV-L20R RS485 の時

端子台



(切替 SW を 485(2)側にする)

シールドは片側を
コネクタハウジングに
接続または接地

※ [詳細は、各メーカーの取扱説明書参照]

図 3.6

3.4 通信を行うまでの各種設定

準備

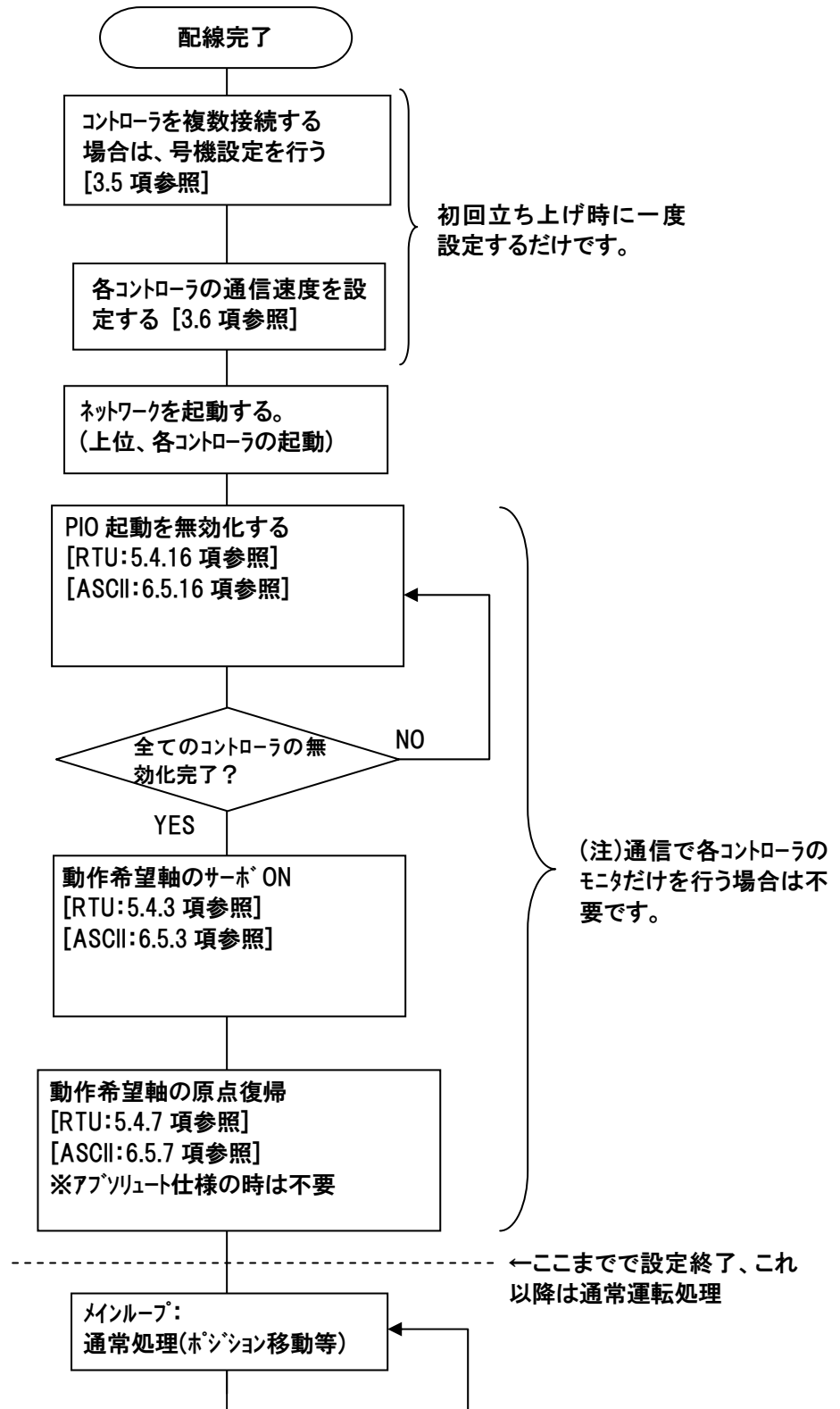


図 3.7

3.5 軸番号の設定

SIO リンク上の各 RC コントローラに 16 軸目を F_H とした $0 \sim F_H$ の 16 進数で軸番号を設定します。

RC コントローラのパネル面に軸番号設定 SW (ADRS) が有る場合 (PCON-C/CG/CF/CA/CFA、ACON-C/CG、SCON-C/CA、ROBONET) は、マイナスドライバーで軸番号に矢印を合わせてください。(軸番号を重複させないでください。)



図 3.8

軸番号設定用 SW が無い RC コントローラは、パソコン対応ソフトなどのティーチングツールで設定します。ここでは、パソコン対応ソフトからの設定を紹介します。[ティーチングボックスからの設定は、それぞれの取扱説明書 (CON-PTA、CON-PT、CON-T、RCM-E、RCM-T) を参照]

パソコンを軸番号設定したい RC コントローラの SIO コネクタに接続します。

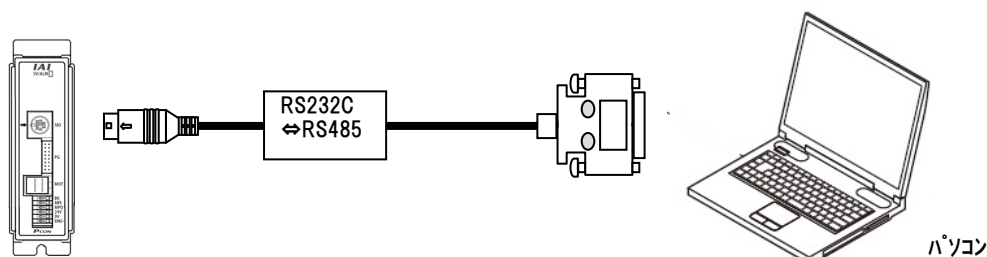


図 3.9

以下の手順で設定を行ってください。

- ① RC 用パソコン対応ソフト起動し、設定(S)をクリック→②コントローラ設定(C)にカーソルを合わせます。
- ③軸番号割付(N)にカーソルを合わせクリック→④軸番号テーブルに、番号が重ならないようにして軸番号(0~15)を入力します。

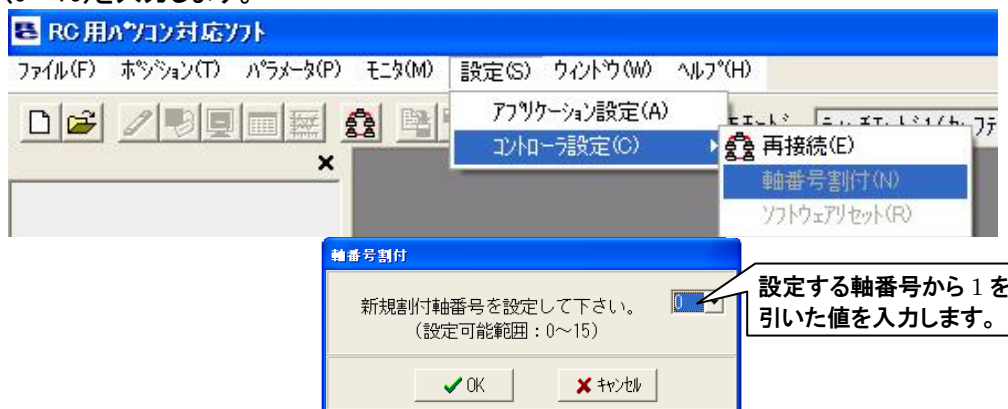


図 3.10

3.6 コントローラの通信速度を設定

通信を行う場合、PLC、各 RC コントローラの通信速度を合わせる必要があります。

通信速度は、3.6.1 項、3.6.2 項の順に設定してください。[上位側の設定は、上位の取扱説明書を参照]

システム構成に応じて配線が異なりますので、ご注意ください。

3.6.1 各システム毎の配線およびハードウェアの設定

(1) 上位(ホスト)コントローラにパソコンを使用する場合

現在の接続のまま設定を行うことができます。モード切替 SW のある RC コントローラ (PCON-C/CG/CF/CA/CFA、ACON-C/CG、SCON-C/CA)は、モード切替 SW を MANU に設定してください。

(2) 上位(ホスト)コントローラに PLC を使用して RS232C で接続している場合

PLC の代わりに上位(ホスト)コントローラにパソコンを接続してください(図 3.1 参照)。その際、PLC と SIO 変換器の接続を外し、RC 用パソコン対応ソフト付属のケーブルを使用して SIO 変換器のティーチングポート(ミニ DIN8 ピンコネクタ)[3.1(3)項参照]に接続してください。モード切替 SW のある RC コントローラ (PCON-C/CG/CF/CA/CFA、ACON-C/CG、SCON-C/CA)は、モード切替 SW を MANU に設定してください。

(3) 上位(ホスト)コントローラに PLC を使用して RS485 で接続している場合

軸番号の設定と同様に各 RC コントローラに直接パソコンを接続してください。モード切替 SW のある RC コントローラ(PCON-C/CG/CF/CA/CFA、ACON-C/CG、SCON-C/CA)は、モード切替 SW を MANU に設定してください。

(4) ROBONET を接続している場合

ROBONET の設定を行う場合は、ゲートウェイユニットのティーチングポートにパソコン対応ソフト付属ケーブルを接続してください。GateWayR ユニットの MODE 切替 SW を MANU に設定してください。

3.6.2 通信速度の設定

以下の手順で設定を行ってください。

(注)ROBONET は、ROBONET ゲートウェイパラメータ設定ツールで設定します。[詳細は、別冊 ROBONET 取扱説明書参照]

- ① RC 用パソコン対応ソフト起動し、パラメータ(P)→編集(E)をクリックします

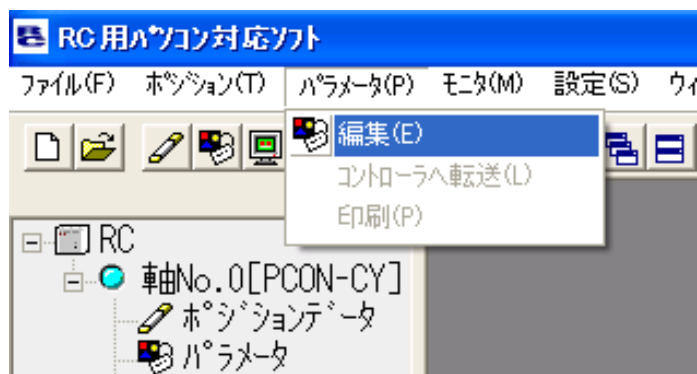


図 3.11

②変更を行うコントローラの軸番号を選択します。

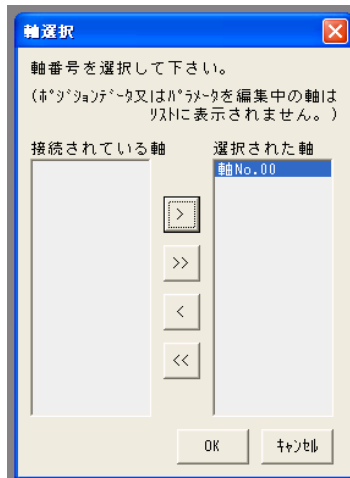


図 3.12

③パラメータNo.16 の SIO 通信速度を設定してください。

No	パラメータ名称	設定値
1	ゾン境界1+側[mm]	100.00
2	ゾン境界1-側[mm]	-0.30
3	ソフトリミット+側[mm]	150.30
4	ソフトリミット-側[mm]	-0.30
5	原点復帰方向[0:逆/1:正]	1
6	押付け停止判定時間[msec]	255
7	サーボゲイン番号	8
8	速度初期値[mm/sec]	600
9	加減速度初期値[G]	0.30
10	位置決め幅初期値[mm]	0.10
11	(将来の拡張のための予約)	0
12	位置決め停止時電流制限値[%]	35
13	原点復帰時電流制限値[%]	35
14	(将来の拡張のための予約)	0
15	(将来の拡張のための予約)	1
16	SIO通信速度[bps]	38400
17	従局トランスミッタ活性化最小遅延時間(RTIM)[msec]	5

図 3.13

4 通信

4.1 メッセージ送信タイミング

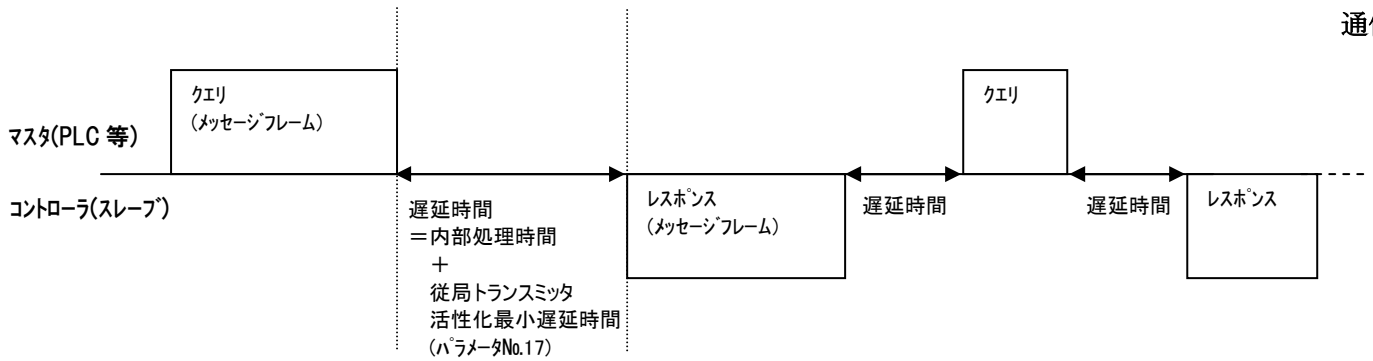


図 4.1

基本的な伝送制御手順は、マスタからのクエリの送信と、これを受信した RC コントローラからのレスポンスの返信を 1 単位とした伝送となります。

受信後、送信までの遅延時間は、「パラメータNo.17「従局トランスミッタ活性化最小遅延時間(初期値 5ms)」と内部処理時間(下表参照)の合計になります。

RC コントローラはクエリ・メッセージ受信完了後、この遅延時間以上経過してからレスポンス・メッセージの送信を開始します。マスタは、クエリ・メッセージ送信完了後、この遅延時間以内に自局を受信可能な状態にする必要があります。

RC コントローラはレスポンス・メッセージの送信完了後、直ちに次のクエリ受信に備えます。

内部処理時間^(注 1)

項目	時間
低速メモリ領域以外の読出し／書込み	最大 1 msec
ポジションデータ(1 ポジション)読出し	最大 4 msec
ポジションデータ(1 ポジション)書込み	最大 15 msec
ポジションデータ(1 ポジション)読出し／書込み	最大 18 msec
ポジションデータ(9 ポジション)読出し	最大 9 msec
ポジションデータ(9 ポジション)書込み	最大 90 msec
ポジションデータ(9 ポジション)読出し／書込み	最大 98 msec

注 1 アクセスするカテゴリおよびコントローラの種類によって処理時間が異なります

4.2 タイムアウトとリトライ

上位ホストはクエリ送信完了後、コントローラからのレスポンスを待ちます。(ブロードキャストクエリの場合を除く)

コマンド送信後、レスポンス受信完了までの経過時間がタイムアウト値 Tout を超えた場合、上位ホストはコマンド再送による通信回復を行うように設定して下さい。リトライ回数が 3 回を超えた場合は、回復不可能な通信異常としてください。

以下にタイムアウト値 Tout 算出方法を示します。

1. タイムアウト値 Tout


$$\text{Tout} = \text{To} + \alpha + (10 \times \text{Bprt} / \text{Kbr}) \text{ [ms]}$$

To : 内部処理時間※ × 安全率 3

α : 従局トランスミッタ活性化最小遅延時間 [ms](パラメータNo.17 初期値 5 ms)

Kbr : 通信速度 [kbps]

Bprt: レスポンス・メッセージのバイト数+8

 **注意** 内部処理時間は、アクセスするカテゴリおよびコントローラの種類によって処理時間が異なります。以下の表に処理時間を示します。

項目	最大時間[ms]
低速メモリ領域以外の読出し／書込み	1
ポジションデータ(1 ポジション)読出し	4
ポジションデータ(1 ポジション)書込み	15
ポジションデータ(1 ポジション)読出し／書込み	18
ポジションデータ(9 ポジション)読出し	9
ポジションデータ(9 ポジション)書込み	90
ポジションデータ(9 ポジション)読出し／書込み	98

2. リトライ回数

Nrt = 3(リトライは 必ず設定して下さい。)

4.3 RCコントローラの内部アドレス および データ構造

RCコントローラのメモリ領域は、ワード単位で読み書きを行う Modbus レジスタ領域と、ビット(コイル)単位で書き込みを行う Modbus ステータス領域によって構成されています。

メモリ領域	アクセス単位	アドレス範囲	ファンクション	
			コード ^(注)	機能
Modbus レジスタ [4.3.1、4.3.2 項 参照]	ワード	0500~9908 _H	03 _H	保持レジスタ読出し
			06 _H	保持レジスタへの書き込み
			10 _H	複数保持レジスタへの一括書き込み
Modbus ステータス [4.3.3、4.3.4 項 参照]	ビット	0100~043F _H	05 _H	コイルへの書き込み

(注)本書で説明しているファンクションコードを示します。

4.3.1 Modbus レジスタの構造

Modbus レジスタは次のように配置されています。

0000 _H	(システム用) ^(注)
0500 _H	最後に検出したアラームの詳細情報
0505 _H	
0D00 _H	(システム用) ^(注)
0D03 _H	I/O 制御情報カテゴリ
1000 _H	(システム用) ^(注)
3FFF _H	ポジションテーブル情報 《低速メモリ領域》
8400 _H	(システム用) ^(注)
842E _H	メンテナンス情報 ※対応機種は、PCON-CA/CF A, SCON-CA、ERC3
9000 _H	(システム用) ^(注)
9015 _H	コントローラモニタ情報カテゴリ
9800 _H	(システム用) ^(注)
9900 _H	ポジション指令カテゴリ
9908 _H	(システム用) ^(注)
9908 _H	直値指令カテゴリ
FFFF _H	(システム用) ^(注)

注 システム用領域は通信に使用できません。

4. 3. 2 Modbus レジスタ詳細

アドレス[HEX]	エリア名称	内容	記号	掲載頁						
				RTU		ASCII				
0000~04FF	システム用									
0500	最後に検出したアラームの詳細情報	アラーム詳細コード*	ALA0	64	30	192	30			
0501		アラームアドレス	ALA0		30		30			
0502		常に 0	—		—		—			
0503		アラームコード*	ALC0		31		31			
0504		アラーム発生時刻	ALT0		32		32			
0506~0CFF	システム用									
0D00	I/O 制御情報カテゴリ	デバイス制御レジスタ 1	DRG1	153	33	285	33			
0D01		デバイス制御レジスタ 2	DRG2		34		34			
0D03		ポジション番号指定レジスタ	POSR		35		35			
0D04~0FFF	システム用									
1000~3FFF	ポジションテーブル情報 (低速メモリ領域)	オフセット[HEX]								
		+0000 _H	目標位置	PCMD	175	177	307	309		
		+0002 _H	位置決め幅	INP						
		+0004 _H	速度指令	VCMD						
		+0006 _H	個別ゾーン境界+側	ZNMP						
		+0008 _H	個別ゾーン境界-側	ZNLP						
		+000A _H	加速度指令	ACMD					178	310
		+000B _H	減速度指令	DCMD						
		+000C _H	押付け時電流制限値	PPOW						
		+000D _H	負荷電流閾値	LPOW						
+000E _H	制御フラグ指定	CTLF								
※ 詳細アドレスの計算は右の通り → ※ アドレス = 1000 _H + (16 × ポジションNo.) + オフセット										
4000~83FF	システム用									
8400	メンテナンス情報 (カレンダー機能対応機種専用)	通算移動回数 ^(注1)	TLMC	36	66	36	194			
8402		通算走行距離 ^(注1)	ODOM	37	68	37	196			
841A		現在時刻(SCON-CA 専用)	TIMN	38	70	38	198			
8420		現在時刻(PCON-CA/CFA 専用)	TIMN	38	70	38	198			
842E		FAN 通算駆動時間(PCON-CFA 専用)	TFAN	39	74	39	202			
8430~8FFF	システム用									
9000	コントローラモニタ情報カテゴリ	現在位置レジスタ	PNOW	(61)	80	(189)	212			
9002		現在発生アラームコードレジスタ	ALMC		82		214			
9003		I/O ポート(入力)レジスタ	DIPM		84		216			
9004		I/O ポート(出力)レジスタ	DOPM		88		220			
9005		デバイスステータス 1 レジスタ	DSS1	40(61)	92	40(189)	224			
9006		デバイスステータス 2 レジスタ	DSS2	41(61)	94	41(189)	226			
9007		拡張デバイスレジスタ	DSSE	42(61)	96	42(189)	228			
9008		システムステータスレジスタ	STAT	43(61)	98	43(189)	230			
900A		現在速度モニタレジスタ	VNOW	(61)	100	(189)	232			
900C		電流値モニタレジスタ	CNOW		102		234			
900E		偏差モニタレジスタ	DEVI		104		236			
9010		システムタイマレジスタ	STIM		106		238			
9012		特殊入力ポートレジスタ	SIPM		44(61)		108	44(189)	240	
9013		ゾーンステータスレジスタ	ZONS	45(61)	110	45(189)	242			
9014		位置決め完了ポジションNo.レジスタ	POSS	46(61)	112	46(189)	244			
9015	拡張システムステータスレジスタ	SSSE	47(61)	110	47(189)	238				

注1 PCON-CA/CFA、SCON-CA、ERC3 専用

アドレス[HEX]	エリア名称	内容	記号	掲載頁			
				RTU		ASCII	
9016~901D	システム用						
901E	コントローラ モニタ情報 カテゴリ	カフィードバックデータ(SCON-CA 専用)	FBFC	(61)	112	(189)	240
901F~97FF	システム用						
9800	ポジション 指令カテゴリ	ポジション移動指令レジスタ	POSR	35	149	35	276
9801~98FF	システム用						
9900	直値指令 カテゴリ	目標位置指定レジスタ	PCMD	157	159	289	291
9902		位置決め幅指定レジスタ	INP				
9904		速度指定レジスタ	VCMD				
9906		加減速度指定レジスタ	ACMD	160	292		
9907		押付け時電流制限指定レジスタ	PPOW				
9908		制御フラグ指定レジスタ	CTLF	161	293		
9909~FFFF	システム用						

通信

(1) アラーム詳細コードの内容(アドレス= 0500_H)(ALA0)

ビット	記号	名称	機能
15	—	アラーム詳細コード 32768	アラーム詳細コード番号を示します。 アラーム詳細コードがあるアラームが発生すると出力されます。アラームが発生していない、またはアラーム詳細コードがないアラームの場合は 0 _H です。 アラーム詳細コードは、バイナリコードで読み出されます。 アラーム詳細コードの内容は、アラームコードと併せてコントローラの取扱説明書で確認してください。
14	—	アラーム詳細コード 16384	
13	—	アラーム詳細コード 8192	
12	—	アラーム詳細コード 4096	
11	—	アラーム詳細コード 2048	
10	—	アラーム詳細コード 1024	
9	—	アラーム詳細コード 512	
8	—	アラーム詳細コード 256	
7	—	アラーム詳細コード 128	
6	—	アラーム詳細コード 64	
5	—	アラーム詳細コード 32	
4	—	アラーム詳細コード 16	
3	—	アラーム詳細コード 8	
2	—	アラーム詳細コード 4	
1	—	アラーム詳細コード 2	
0	—	アラーム詳細コード 1	

(2) アラームアドレスの内容(アドレス= 0501_H)(ALA0)

ビット	記号	名称	機能
15	—	アラームアドレス 32768	アラームアドレスを示します。 仮想領域に格納している値が発生したアラームの要因である場合、格納している仮想アドレスを出力します。アラームが発生していない、または仮想領域に原因がないアラームの場合は FFFF _H です。 アラームアドレスは、バイナリコードで読み出されます。
14	—	アラームアドレス 16384	
13	—	アラームアドレス 8192	
12	—	アラームアドレス 4096	
11	—	アラームアドレス 2048	
10	—	アラームアドレス 1024	
9	—	アラームアドレス 512	
8	—	アラームアドレス 256	
7	—	アラームアドレス 128	
6	—	アラームアドレス 64	
5	—	アラームアドレス 32	
4	—	アラームアドレス 16	
3	—	アラームアドレス 8	
2	—	アラームアドレス 4	
1	—	アラームアドレス 2	
0	—	アラームアドレス 1	

(3) アラームコードの内容(アドレス= 0503_H)(ALCO)

ビット	記号	名称	機能
15	—	アラームコード [°] 32768	<p>各レベル(コールトスタート、動作解除、メッセージ)のアラームコード番号を示します。</p> <p>アラームが発生すると出力されます。アラームが発生していない場合は 0_Hです。</p> <p>アラームコードは、バイナリコードで読み出されます。</p> <p>アラームコードの内容は、コントローラの取扱説明書で確認してください。</p> <p>(注)メッセージレベルアラームは、発生しないコントローラがあります。[詳細は、各コントローラのトラブルシューティングを参照]</p> <p>(参考)メッセージレベルアラームが発生しないコントローラから、発生するコントローラに置き換えた場合(例:PCON-C⇒PCON-CA)、メッセージレベルアラーム発生時の動作を検討してください。</p>
14	—	アラームコード [°] 16384	
13	—	アラームコード [°] 8192	
12	—	アラームコード [°] 4096	
11	—	アラームコード [°] 2048	
10	—	アラームコード [°] 1024	
9	—	アラームコード [°] 512	
8	—	アラームコード [°] 256	
7	—	アラームコード [°] 128	
6	—	アラームコード [°] 64	
5	—	アラームコード [°] 32	
4	—	アラームコード [°] 16	
3	—	アラームコード [°] 8	
2	—	アラームコード [°] 4	
1	—	アラームコード [°] 2	
0	—	アラームコード [°] 1	

注 アドレス=0502_Hは、常に 0 を返します。

(4) アラーム発生時刻の内容(アドレス= 0504_H)(ALTO)

ビット	記号	名称	機能
31	—	アラーム発生時刻 2147202832	<p>アラームの発生した時刻または時間を出力します。</p> <p>①カレンダー機能(RTC)を搭載した機種で、RTCを有効に設定している場合、アラーム発生時刻になります。</p> <p>②RTCを無効に設定、またはRTCが無い機種の場合、コントローラ電源投入からの経過時間[sec]になります。</p> <p>●①のアラーム発生時刻の計算方法</p> <p>アラーム発生時刻のデータは、基準時刻(2000年1月1日00時00分00秒)からの経過秒を示します。</p> <p>基準時刻からの経過秒をS、経過分をM、経過時をH、経過日をD、経過年をYとし、以下の式で計算を行います。</p> <p>S= 読み込んだアラーム発生時刻のデータ M= S/60(小数点以下切捨て) H= M/60(小数点以下切捨て) D= H/24(小数点以下切捨て) Y= D/365.25(小数点以下切捨て) L(閏年計算)= Y/4(小数点切り上げ)</p> <p>アラーム発生時刻の秒をSA、分をMA、時をHA、今年になってからの経過日をDA、年をYAとすると、以下の式で時刻を計算できます。</p> <p>SA= S/60の余り MA= M/60の余り HA= H/24の余り DA= D-(Y×365+L) DAから一月ごとの日数を減算することで月日を求めます。 YA= Y+2000(西暦)</p> <p>例) アラーム発生時刻のデータが172C1B8B_Hだった場合 (1)10進数に変換: S= 172C1B8B_H⇒388766603 (2)M、H、D、Y、Lを計算します。 M= 388766603/60= 6479443 H= 6479443/60= 107990 D= 107990/24= 4499 Y= 4499/365.25= 12 L= 12/4= 3 (3)SA、MA、HA、およびDAを求めます。 SA= 388766603/60の余り= 23 MA= 6479443/60の余り= 43 HA= 107990/24の余り= 14 DA= 4499-(12×365+3) = 116(今年になって116日経過し、アラーム発生時は117日目) 月日= 117-{31(1月)-29(2月)-31(3月)}= 26(4月分を減算すると負数になってしまうので、アラーム発生時は4月26日) YA= 12+2000= 2012 以上より、アラーム発生時刻は、2012年4月26日14時43分23秒となります。</p>
30	—	アラーム発生時刻 1073601416	
29	—	アラーム発生時刻 536800708	
28	—	アラーム発生時刻 268400354	
27	—	アラーム発生時刻 134200177	
26	—	アラーム発生時刻 67108864	
25	—	アラーム発生時刻 33554432	
24	—	アラーム発生時刻 16777216	
23	—	アラーム発生時刻 8388608	
22	—	アラーム発生時刻 4194304	
21	—	アラーム発生時刻 2097152	
20	—	アラーム発生時刻 1048576	
19	—	アラーム発生時刻 524288	
18	—	アラーム発生時刻 262144	
17	—	アラーム発生時刻 131072	
16	—	アラーム発生時刻 65536	
15	—	アラーム発生時刻 32768	
14	—	アラーム発生時刻 16384	
13	—	アラーム発生時刻 8192	
12	—	アラーム発生時刻 4096	
11	—	アラーム発生時刻 2048	
10	—	アラーム発生時刻 1024	
9	—	アラーム発生時刻 512	
8	—	アラーム発生時刻 256	
7	—	アラーム発生時刻 128	
6	—	アラーム発生時刻 64	
5	—	アラーム発生時刻 32	
4	—	アラーム発生時刻 16	
3	—	アラーム発生時刻 8	
2	—	アラーム発生時刻 4	
1	—	アラーム発生時刻 2	
0	—	アラーム発生時刻 1	

(5) デバイス制御レジスタ1内容(アドレス= 0D00_H)(DRG1)

ビット	記号	名称	機能
15	EMG	EMG 動作指定	0:非常停止解除状態 1:非常停止 本ビットを1にすると非常停止状態となります。ただし駆動源しゃ断は行ないません。 (コントローラ本体の ALM LED は点灯しません)
14	SFTY	セーフティ速度指令	0:セーフティ速度無効 1:セーフティ速度有効 本ビットを1にすると、No.35 パラメータの「セーフティ速度」で指定された速度により、全ての移動指令速度が制限されます。
13	—	使用できません	
12	SON	サーボ ON 指令	0:サーボ OFF 1:サーボ ON 本ビットを1にするとサーボ ON 状態となります。 ただし次の条件を満たしている必要があります。 ・デバイスステータスレジスタ 1(5.3.11 項または 6.4.11 項) :EMG ステータスビット=0 ・デバイスステータスレジスタ 1(5.3.11 項または 6.4.11 項) :重故障ステータスビット=0 ・デバイスステータスレジスタ 2(5.3.12 項または 6.4.12 項) :イネーブルステータスビット=1 ・システムステータスレジスタ(5.3.9 項または 6.4.9 項) :自動サーボ OFF 中ステータスビット=0
11~9	—	使用できません	
8	ALRS	アラームリセット指令	0:通常 1:アラームリセット実行 本ビットへの立上がりエッジ入力(0→1)で、アラームがリセットされます。ただしアラーム要因が解消されていないと再びアラームとなります。また一時停止中に本ビットに立上がりエッジ入力(0→1)されると残移動量のキャンセルが行われます。
7	BKRL	ブレーキ強制解除指令	0:通常 1:ブレーキ強制解除 本ビットを1にすると、ブレーキを強制的に解除することができます。
6	—	使用できません	
5	STP	一時停止指令	0:通常 1:一時停止指令 本ビットが1の状態では、モータの移動は全て禁止されます。移動中に本ビットが1になった場合は減速停止します。再び0にセットされると残り分の移動を行います。 原点復帰中に本ビットを1にした場合、押付け反転前は移動指令が保留され、0になると自動で残りの原点復帰動作を行います。押付け反転後では、再度原点復帰を行ってください。
4	HOME	原点復帰指令	0:通常 1:原点復帰指令 本ビットへの立上がりエッジ入力(0→1)で、原点復帰動作を行います。原点復帰が完了すると HEND ビットが1になります。 原点復帰指令は、原点復帰が完了済みでも再度入力可能です。
3	CSTR	位置決め動作起動指令	0:通常 1:ポジションスタート指令 本ビットへの立上がりエッジ入力(0→1)で、ポジション番号指定レジスタ (POSR:0D03 _H)内のポジション番号の指定位置に移動します。本ビットが1のままでは位置決め幅内に入っても完了ポジションは出力されません。(0 を書き込んで通常状態に戻してください。) 電源投入後、一度も原点復帰動作を行っていない状態(HEND=0)では、原点復帰動作を実行後、目標位置に移動します。 ※目標位置および速度等は、コントローラ内部のポジションテーブルに予め設定しておいてください。
2~0	—	使用できません	

(6) デバイス制御レジスタ 2 内容(アドレス= 0D01_H)(DRG2)

ビット	記号	名称	機能
15	—	使用できません	
14	JISL	ジヨグ/インチング切替え	0:ジヨグ 1:インチング 0 の時、ジヨグ動作が選択され、1 の時はインチング動作が選択されます。 ジヨグ動作中に本ビットが 1 になると減速停止します。 インチング動作中に本ビットが 0 になってもインチング動作を継続します。 本ビットの設定は、ティーチングツールのジヨグ/インチング動作には反映されません。
13	—	使用できません	
12	—	使用できません	
11	MOD	ティーチモード指令	0:通常運転モード 1:教示モード 本ビットを 1 にすると教示モードになります。
10	TEAC	ポジションデータ取込み指令	0:通常 1:ポジションデータ取込み指令 11ビット目のティーチモード指令が 1(教示モード)の時に、本ビットに 1 を書き込むことで現在位置データの取込みを行います。 取込み場所は、ポジション番号指定レジスタで指定されているポジションデータの中です。取込みポジションが、データが設定されていない空のポジションの場合、目標位置以外のデータ(位置決め幅等)はパラメータの初期値が書込まれます。 本ビットに 1 をセットしたら、20 msec 以上そのままの状態を保ってください。
9	JOG+	ジヨグ+指令	0:通常 1:ジヨグ+指令 ・14ビット目の JISL ビットが 0 の時、本ビットを 1 にしている間、反原点方向にジヨグ移動を行います。速度および加減速度はユーザパラメータ No.26 の PIO ジヨグ速度と定格加減速度が使用されます。 ジヨグ移動中に本ビットを 0 にするか、8ビット目のジヨグ-指令を 1 にすると減速停止します。 ・14ビット目の JISL ビットが 1 の時、ジヨグ+指令のエッジを立てる(本ビット:0→1)と、反原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータ No.26(PIO ジヨグ速度)、ユーザパラメータ No.48(PIO インチング距離)、定格加減速度が使用されます。
8	JOG-	ジヨグ-指令	0:通常 1:ジヨグ-指令 ・14ビット目の JISL ビットが 0 の時、本ビットを 1 にしている間、原点方向にジヨグ移動を行います。速度および加減速度はユーザパラメータ No.26 の PIO ジヨグ速度と定格加減速度が使用されます。 ジヨグ移動中に本ビットを 0 にするか、9ビット目のジヨグ+指令を 1 にすると減速停止します。 ・14ビット目の JISL ビットが 1 の時、ジヨグ-指令のエッジを立てる(本ビット:0→1)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータ No.26(PIO ジヨグ速度)、ユーザパラメータ No.48(PIO インチング距離)、定格加減速度が使用されます。
7	ST7	スタートポジション 7	指定されたポジション No.位置に移動します。
6	ST6	スタートポジション 6	PIO パターン 4、5(電磁弁モード)が選択されている時に使用できます。
5	ST5	スタートポジション 5	ST0～ST7 のどれかを 1 にする(本ビット:0→1)と移動を行います。
4	ST4	スタートポジション 4	有効スタートポジション以外を選択するとアラーム「085 移動時ポジション No.異常」が発生します。
3	ST3	スタートポジション 3	
2	ST2	スタートポジション 2	ユーザパラメータ No.27 移動指令種別で、信号入力方法をレベルとエッジを選択可能です。
1	ST1	スタートポジション 1	
0	ST0	スタートポジション 0	同時に複数のポジションを入力すると若い番号が優先されます。

(7) ポジション番号指定レジスタ内容(アドレス= 0D03_H)(POSR)ポジション移動指定レジスタ内容(アドレス= 9800_H)(POSR)

ビット	記号	名称	機能
15	—	使用できません	
14	—	使用できません	
13	—	使用できません	
12	—	使用できません	
11	—	使用できません	
10	—	使用できません	
9	PC512	ポジション指定ビット 512	<p>移動したいポジションNo.をバイナリコードで指定します。 機種や、PIO パターンにより最大ポジション数は異なりますのでご 注意ください。</p> <p>【アドレス=0D03_Hを使用する場合】 ポジションNo.指定の後、デバイス制御レジスタ1のCSTR(スタート信 号)を1にするとポジション移動します。[5.5.1 項または 6.6.1 項参照]</p> <p>【アドレス=9800_Hを使用する場合】 本レジスタは、ポジションNo.が指定されると移動を行います。 CSTR(スタート信号)は必要ありません。</p>
8	PC256	ポジション指定ビット 256	
7	PC128	ポジション指定ビット 128	
6	PC64	ポジション指定ビット 64	
5	PC32	ポジション指定ビット 32	
4	PC16	ポジション指定ビット 16	
3	PC8	ポジション指定ビット 8	
2	PC4	ポジション指定ビット 4	
1	PC2	ポジション指定ビット 2	
0	PC1	ポジション指定ビット 1	

通信

(8) 通算移動回数の内容(アドレス= 8400_H)(TLMC)

ビット	記号	名称	機能
31	—	通算移動回数 2147202832	通算移動回数を示します。 通算移動回数は、バイナリコードで読み出されます。 ※対応機種:PCON-CA/CFA、SCON-CA、ERC3
30	—	通算移動回数 1073601416	
29	—	通算移動回数 536800708	
28	—	通算移動回数 268400354	
27	—	通算移動回数 134200177	
26	—	通算移動回数 67108864	
25	—	通算移動回数 33554432	
24	—	通算移動回数 16777216	
23	—	通算移動回数 8388608	
22	—	通算移動回数 4194304	
21	—	通算移動回数 2097152	
20	—	通算移動回数 1048576	
19	—	通算移動回数 524288	
18	—	通算移動回数 262144	
17	—	通算移動回数 131072	
16	—	通算移動回数 65536	
15	—	通算移動回数 32768	
14	—	通算移動回数 16384	
13	—	通算移動回数 8192	
12	—	通算移動回数 4096	
11	—	通算移動回数 2048	
10	—	通算移動回数 1024	
9	—	通算移動回数 512	
8	—	通算移動回数 256	
7	—	通算移動回数 128	
6	—	通算移動回数 64	
5	—	通算移動回数 32	
4	—	通算移動回数 16	
3	—	通算移動回数 8	
2	—	通算移動回数 4	
1	—	通算移動回数 2	
0	—	通算移動回数 1	

(9) 通算走行距離の内容(アドレス= 8402_H)(ODOM)

ビット	記号	名称	機能
31	—	通算走行距離 2147202832	通算走行距離[m]を示します。 通算走行距離は、ハイナリコードで読み出されます。 ※対応機種:PCON-CA/CFA、SCON-CA、ERC3
30	—	通算走行距離 1073601416	
29	—	通算走行距離 536800708	
28	—	通算走行距離 268400354	
27	—	通算走行距離 134200177	
26	—	通算走行距離 67108864	
25	—	通算走行距離 33554432	
24	—	通算走行距離 16777216	
23	—	通算走行距離 8388608	
22	—	通算走行距離 4194304	
21	—	通算走行距離 2097152	
20	—	通算走行距離 1048576	
19	—	通算走行距離 524288	
18	—	通算走行距離 262144	
17	—	通算走行距離 131072	
16	—	通算走行距離 65536	
15	—	通算走行距離 32768	
14	—	通算走行距離 16384	
13	—	通算走行距離 8192	
12	—	通算走行距離 4096	
11	—	通算走行距離 2048	
10	—	通算走行距離 1024	
9	—	通算走行距離 512	
8	—	通算走行距離 256	
7	—	通算走行距離 128	
6	—	通算走行距離 64	
5	—	通算走行距離 32	
4	—	通算走行距離 16	
3	—	通算走行距離 8	
2	—	通算走行距離 4	
1	—	通算走行距離 2	
0	—	通算走行距離 1	

通信

(10) 現在時刻の内容(アドレス = 841A_H(SCON-CA)、8420_H(PCON-CA/CFA)(TIMN))

ビット	記号	名称	機能
31	—	現在時刻 2147202832	<p>現在の時刻または時間を出力します。</p> <p>①カレンダー機能(RTC)を搭載した機種で、RTCを有効に設定している場合、現在時刻になります。</p> <p>②RTCを無効に設定、またはRTCが無い機種の場合、コントローラ電源投入からの経過時間[sec]になります。</p> <p>●①の現在時刻の計算方法</p> <p>現在時刻のデータは、基準時刻(2000年1月1日00時00分00秒)からの経過秒を示します。</p> <p>基準時刻からの経過秒をS、経過分をM、経過時をH、経過日をD、経過年をYとし、以下の式で計算を行います。</p> <p>S= 読込んだ現在時刻のデータ M= S/60(小数点以下切捨て) H= M/60(小数点以下切捨て) D= H/24(小数点以下切捨て) Y= D/365.25(小数点以下切捨て) L(閏年計算)= Y/4(小数点切り上げ)</p> <p>時刻の秒をSA、分をMA、時をHA、今年になってからの経過日をDA、年をYAとすると、以下の式で時刻を計算できます。</p> <p>SA= S/60の余り MA= M/60の余り HA= H/24の余り DA= D-(Y×365+L) DAから一月ごとの日数を減算することで月日を求めます。 YA= Y+2000 (西暦)</p> <p>例) 現在時刻のデータが172C1B8B_Hだった場合 (1)10進数に変換: S= 172C1B8B_H⇒388766603_D (2)M、H、D、Y、Lを計算します。 M= 388766603/60= 6479443 H= 6479443/60= 107990 D= 107990/24= 4499 Y= 4499/365.25= 12 L= 12/4= 3 (3)SA、MA、HA、およびDAを求めます。 SA= 388766603/60の余り= 23 MA= 6479443/60の余り= 43 HA= 107990/24の余り= 14 DA= 4499-(12×365+3) = 116(今年になって116日経過し、現在は117日目) 月日= 117-{31(1月)-29(2月)-31(3月)}= 26(4月分を減算すると負数になってしまうので、発生時は4月26日) YA= 12+2000= 2012 以上より、現在時刻は、2012年4月26日14時43分23秒となります。</p>
30	—	現在時刻 1073601416	
29	—	現在時刻 536800708	
28	—	現在時刻 268400354	
27	—	現在時刻 134200177	
26	—	現在時刻 67108864	
25	—	現在時刻 33554432	
24	—	現在時刻 16777216	
23	—	現在時刻 8388608	
22	—	現在時刻 4194304	
21	—	現在時刻 2097152	
20	—	現在時刻 1048576	
19	—	現在時刻 524288	
18	—	現在時刻 262144	
17	—	現在時刻 131072	
16	—	現在時刻 65536	
15	—	現在時刻 32768	
14	—	現在時刻 16384	
13	—	現在時刻 8192	
12	—	現在時刻 4096	
11	—	現在時刻 2048	
10	—	現在時刻 1024	
9	—	現在時刻 512	
8	—	現在時刻 256	
7	—	現在時刻 128	
6	—	現在時刻 64	
5	—	現在時刻 32	
4	—	現在時刻 16	
3	—	現在時刻 8	
2	—	現在時刻 4	
1	—	現在時刻 2	
0	—	現在時刻 1	

(11) FAN 通算駆動時間の内容(アドレス = 842E_H)(TFAN)

ビット	記号	名称	機能
31	—	FAN 通算駆動時間 2147202832	FAN 通算駆動時間[sec]を示します。 FAN 通算駆動時間は、バイナリコードで読み出されます。 ※対応機種:PCON-CFA
30	—	FAN 通算駆動時間 1073601416	
29	—	FAN 通算駆動時間 536800708	
28	—	FAN 通算駆動時間 268400354	
27	—	FAN 通算駆動時間 134200177	
26	—	FAN 通算駆動時間 67108864	
25	—	FAN 通算駆動時間 33554432	
24	—	FAN 通算駆動時間 16777216	
23	—	FAN 通算駆動時間 8388608	
22	—	FAN 通算駆動時間 4194304	
21	—	FAN 通算駆動時間 2097152	
20	—	FAN 通算駆動時間 1048576	
19	—	FAN 通算駆動時間 524288	
18	—	FAN 通算駆動時間 262144	
17	—	FAN 通算駆動時間 131072	
16	—	FAN 通算駆動時間 65536	
15	—	FAN 通算駆動時間 32768	
14	—	FAN 通算駆動時間 16384	
13	—	FAN 通算駆動時間 8192	
12	—	FAN 通算駆動時間 4096	
11	—	FAN 通算駆動時間 2048	
10	—	FAN 通算駆動時間 1024	
9	—	FAN 通算駆動時間 512	
8	—	FAN 通算駆動時間 256	
7	—	FAN 通算駆動時間 128	
6	—	FAN 通算駆動時間 64	
5	—	FAN 通算駆動時間 32	
4	—	FAN 通算駆動時間 16	
3	—	FAN 通算駆動時間 8	
2	—	FAN 通算駆動時間 4	
1	—	FAN 通算駆動時間 2	
0	—	FAN 通算駆動時間 1	

通信

(12) デバイスステータスレジスタ1 内容(アドレス= 9005_H)(DSS1)

ビット	記号	名称	機能
15	EMGS	EMG ステータス	0:非常停止解除状態 1:非常停止状態 非常停止入力および駆動源しゃ断等で、コントローラが非常停止状態であるかを示します。
14	SFTY	セーフティ速度有効ステータス	0:セーフティ速度無効 1:セーフティ速度有効 デバイス制御レジスタ1の「セーフティ速度指令ビット」で、コントローラのセーフティ速度の有効/無効を示します。
13	PWR	コントローラデバイスステータス	0:コントローラ BUSY 1:コントローラ READY コントローラが外部より制御可能かを示します。通常はBUSYになることはありません。
12	SV	サーボ ON ステータス	0:サーボ OFF 1:サーボ ON サーボ ON 状態を示します。サーボ ON 指令後、パラメータに設定したサーボ ON 遅延時間が経過するまで本ビットは0のままです。またサーボ ON 指令を行ってもサーボ ON できない場合も本ビットは0のままです 本ビットが0では、RCコントローラは一切の移動動作指令を受付けません。
11	PSFL	押付空振り	0:通常 1:押付け空振り 押付け動作指令を行い、ワークに当たらずに押付け幅まで移動した時(押付け空振りに)1となります。 押付け動作以外の動作指令では0のままです。
10	ALMH	重故障ステータス	0:通常 1:重故障アラーム発生中 コールドスタートレベル または動作解除レベルのアラーム発生で1となります。 動作解除レベルのアラーム解除はアラームリセット指令により行えますが、コールドスタートレベルは電源再投入が必要です。
9	ALML	軽故障ステータス	0:通常 1:軽故障アラーム発生中 メッセージレベルのアラームが発生した場合に1となります。
8	ABER	アブソエラーステータス	0:通常 1:アブソエラー発生中 アブソリュート仕様の場合、アブソ関連のエラー発生で1になります。
7	BKRL	ブレーキ強制解除ステータス	0:ブレーキ動作中 1:ブレーキ解除中 ブレーキの状態を示します。通常サーボ ON 中は1となります。サーボ OFF 中であってもデバイス制御レジスタ1の「ブレーキ強制解除指令ビット」を1にすることで1となります。
6	—	使用できません	
5	STP	一時停止中ステータス	0:通常 1:一時停止指令中 一時停止指令の入力中は1となります。 5.4.16項または6.5.16項PIO/Modbus切替設定がPIO有効の時は一時停止のPIO信号のモニタを行います(モード切替SWがあるRCコントローラは、AUTO側に設定してください)。Modbusが有効の時は5.4.6項または6.5.6項一時停止指令のモニタを行います。
4	HEND	原点復帰完了ステータス	0:原点復帰未完了 1:原点復帰完了 原点復帰完了で1となります。アブソリュート仕様の場合はアブソリセットが完了していれば起動時から1となります。 本ビットが0の状態では移動指令を行うとアラームになります。
3	PEND	位置決め完了ステータス	0:位置決め未完了 1:位置決め完了 目標位置まで移動して、位置決め幅に入ると1となります。起動時のサーボ ON では、その場に位置決めを行ったことになるので1となります。また、押付け動作中 押付け完了で1となります。
2	CEND	ロードセルキャリブレーション完了	0:キャリブレーション未完了 1:キャリブレーション完了 ロードセルキャリブレーション指令(GLBR)に対して、完了で1となります。
1	CLBS	ロードセルキャリブレーションステータス	0:キャリブレーション未完了 1:キャリブレーション完了 ロードセルキャリブレーション指令の有無に関わらず、キャリブレーション済みで1となります。
0	—	使用できません	

(13) デバイスステータスレジスタ 2 内容(アドレス = 9006_H)(DSS2)

ビット	記号	名称	機能
15	ENBS	イネーブル	0:ディセーブル状態(運転停止、サーボ OFF) 1:イネーブル状態(通常動作) イネーブル機能を搭載した機種に、イネーブル SW(テッドマン SW)を搭載した ティーチングツールを接続した場合、イネーブル SW の状態を示します。 (注)AUTO モード時、またはイネーブル機能を搭載していない機種では 1 に固定です。
14	—	使用できません	
13	LOAD	負荷出力判定ステータス	0:通常 1:負荷出力判定 移動指令時に負荷電流閾値および検定範囲(個別ゾーン境界値)が 設定されている場合、検定範囲内でモータ電流が閾値に達したかを 示します。 次の位置指令を受けるまで現在の値を保持します。
12	TRQS	トルクレベルステータス	0:通常 1:トルクレベル達成 押付け動作中、設定した押付けトルクに電流値が達した時に 1 とな ります。 本ビットはレベルを示しているため電流が変化すれば本ビットの状態も 変化します。
11	MODS	ティーチモードステータス	0:通常運転モード 1:教示モード デバイス制御レジスタ 2 の「ティーチモード指令ビット」により教示モードが選 択されたときに 1 となります。
10	TEAC	ポジションデータ取り込み指令 ステータス	0:通常 1:ポジションデータ取り込み完了 デバイス制御レジスタ 2 の「ポジションデータ取り込み指令ビット」を 1 にする と本ビットは 0 となり、ポジションデータが不揮発性メモリ(EEPROM 等)に正 常に書き込まれると本ビットは 1 となります。
9	JOG+	ジョグ+ステータス	0:通常 1:「ジョグ+」指令中 デバイス制御レジスタ 2 の「ジョグ+指令ビット」が選択されている間 1 とな ります。
8	JOG-	ジョグ-ステータス	0:通常 1:「ジョグ-」指令中 デバイス制御レジスタ 2 の「ジョグ-指令ビット」が選択されている間 1 とな ります。
7	PE7	完了ポジション 7	PIO パターン 4、5(電磁弁モード)時に完了ポジションNo.をバイナリで出力し ます。 ポジション移動指令(デバイス制御レジスタ 2 ST0~ST7)により、ポジション 移動が行われ、目標位置まで移動して、位置決め幅に入ると 1 とな ります。 サーボ OFF では 0 となりますが、再びサーボ ON した場合、指令ポジ ションデータの位置決め幅内であれば本ビットは 1 となります。また押付 け動作で押付け完了した時 あるいは空振りした場合に 1 とな ります。
6	PE6	完了ポジション 6	
5	PE5	完了ポジション 5	
4	PE4	完了ポジション 4	
3	PE3	完了ポジション 3	
2	PE2	完了ポジション 2	
1	PE1	完了ポジション 1	
0	PE0	完了ポジション 0	

(14) 拡張デバイスステータスレジスタ内容(アドレス = 9007_H)(DSSE)

ビット	記号	名称	機能
15	EMGP	非常停止ステータス	0:非常停止入力 OFF 1:非常停止入力 ON 非常停止入力ポートの状態を示します。
14	MPUV	モータ電圧低下ステータス	0:通常 1:モータ駆動源しゃ断中 モータ駆動電源の入力がない場合、本ビットは 1 となります。
13	RMDS	運転モードステータス	0:AUTO モード 1:MANU モード RC コントローラが MANU モードの時に 1 となります。 ただし、運転モード SW 非搭載機種(ERC2、PCON-SE、ACON-SE、PCON-CY、ACON-CY)では常に MANU モードとなります。
12	—	使用できません	
11	GHMS	原点復帰中ステータス	0:通常 1:原点復帰中 原点復帰動作中に 1 となります。それ以外では 0 となります。
10	PUSH	押付け動作中	0:通常 1:押付け動作中 押付け動作中(アプローチ動作は除く)に 1 となります。 次の条件により 0 となります。 1.押付け空振り 2.一時停止 3.次の移動指令 4.サーボ OFF
9	PSNS	励磁検出ステータス	0:励磁検出未完了 1:励磁検出完了 PCON・ERC2、3 シリーズコントローラでは起動後最初のサーボ ON 指令の際、励磁検出動作を行います。励磁検出が完了すると 1 となります。 励磁検出動作を失敗すると 0 のままです。 励磁検出完了後、ソフトウェアリセットを行うと 0 になります。 ACON シリーズコントローラでは起動後最初のサーボ ON 指令にてホールセンサ動作を行い、完了すると 1 となります。 SCON シリーズコントローラでは常に 0 です。
8	PMSS	PIO/Modbus 切替えステータス	0:PIO 指令有効 1:PIO 指令無効 5.4.16 項または 6.5.16 項 PIO/Modbus 切替設定により切替えした結果または現在の状態を示します。
7	—	使用できません	
6	—	使用できません	
5	MOVE	移動中信号	0:停止中 1:移動中 移動中(原点復帰、押付け動作中含む)を示します。一時停止中は 0 となります。
4	—	使用できません	
3	—	使用できません	
2	—	使用できません	
1	—	使用できません	
0	—	使用できません	

(15) システムステータスレジスタ内容(アドレス = 9008_H)(STAT)

ビット	記号	名称	機能
31	BATL	アブソリュートバッテリー電圧低下 (SCON 限定)	0:正常 1:バッテリー電圧低下 アブソリュートバッテリーの電圧が、警告レベル以下になると1になります。 本ビットが1であってもデバイスステータスレジスタ1の重故障ステータスビットが0ならば軸動作は可能です。
30 ~ 18	—	使用できません	
17	ASOF	自動サーボ OFF 中	0:通常 1:自動サーボ OFF 中 RC コントローラのパラメータで「自動サーボ OFF 遅延時間」が設定されていて、位置決め完了後、該当時間が経過し自動サーボ OFF となった場合、1 となります。
16	AEEP	不揮発メモリアクセス中	0:通常 1:不揮発メモリアクセス中 RC コントローラのパラメータ・ポジションテーブル等の読書きで、不揮発メモリにアクセスが開始された時 1 となります。 アクセスが完了するかタイムアウトエラーになった時 0 となります。
15 ~ 5	—	使用できません	
4	RMDS	運転モードステータス	0:AUTO モード 1:MANU モード RC コントローラが MANU モードの時に 1 となります。 ただし、運転モード SW 非搭載機種(ERC2、PCON-SE/CY、ACON-SE/CY)では常に MANU モードとなります。
3	HEND	原点復帰完了状態	0:原点復帰未完了 1:原点復帰完了 原点復帰完了で 1 となります。アブソリュート仕様の場合はアブソリュートが完了していれば起動時から 1 となります。 本ビットが 0 の状態で移動指令を行うとアラームになります。
2	SV	サーボ状態	0:サーボ OFF 1:サーボ ON サーボ ON 状態を示します。サーボ ON 指令後、パラメータに設定したサーボ ON 遅延時間が経過するまで本ビットは 0 のままです。またサーボ ON 指令を行ってもサーボ ON できない場合も本ビットは 0 のままです 本ビットが 0 の状態では、RC コントローラは一切の移動動作指令を受け付けません。
1	SON	サーボ指令状態	0:サーボ OFF 1:サーボ ON サーボ ON/OFF の指令状態を示します。 本ビットは以下の条件を満たしている場合 1 となります。 ・デバイスステータスレジスタ 1 の EMG ステータスビットが 0 [5.3.12 項または 6.4.12 項参照] ・デバイスステータスレジスタ 1 の重故障ステータスビットが 0 [5.3.12 項または 6.4.12 項参照] ・デバイスステータスレジスタ 2 のイネーブルステータスビットが 1 [5.3.13 項または 6.4.13 項参照] ・システムステータスレジスタの自動サーボ OFF 中ステータスビットが 0 [5.3.15 項または 6.4.15 項参照]
0	MPOW	駆動源 ON	0:駆動源しや断中 1:通常 モータ駆動電源の入力がない場合に本ビットは 0 となります。

(16) 特殊ポートモニタレジスタ内容(アドレス= 9012_H)(SIPM)

ビット	記号	名称	機能
15	—	使用できません	
14	NP	指令パルス NP 信号状態	指令パルス NP 信号の状態を示します。
13	—	使用できません	
12	PP	指令パルス PP 信号状態	指令パルス PP 信号の状態を示します。
11	—	使用できません	
10	—	使用できません	
9	—	使用できません	
8	MDSW	モードスイッチ状態	0:AUTO モード 1:MANU モード RC コントローラが MANU モードの時に 1 となります。 ただし、運転モード SW 非搭載機種(ERG2、PCON-SE/CY、 ACON-SE/CY)では常に MANU モードとなります。
7	—	使用できません	
6	—	使用できません	
5	—	使用できません	
4	BLCT	ヘルト切断センサ (SCON に限る)	0:ヘルト切断 1:通常
3	HMCK	原点確認センサモニタ	0:センサ OFF 1:センサ ON 原点確認センサ機能搭載機種では、センサ入力の状態を示します。 非搭載機種では常に 0 です。
2	OT	オーパトラベルセンサモニタ	0:センサ OFF 1:センサ ON エンコーダコネクタ内のオーパトラベルセンサ信号の状態を示します。 非搭載機種では常に 0 です。
1	CREP	クリープセンサモニタ	0:センサ OFF 1:センサ ON エンコーダコネクタ内のクリープセンサ信号の状態を示します。 非搭載機種では常に 0 です。
0	LS	リミットセンサモニタ	0:センサ OFF 1:センサ ON エンコーダコネクタ内の原点センサ信号の状態を示します。 非搭載機種では常に 0 です。

(17) ゾーンステータスレジスタ内容(アドレス= 9013_H)(ZONS)

ビット	記号	名称	機能
15	—	使用できません	
14	LS2	リミットセンサ出力モニタ 2 (PCON-C/CG/CA/CFA、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジション No.2 の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
13	LS1	リミットセンサ出力モニタ 1 (PCON-C/CG/CA/CFA、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジション No.1 の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
12	LS0	リミットセンサ出力モニタ 0 (PCON-C/CG/CA/CFA、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジション No.0 の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
11	—	使用できません	
10	—	使用できません	
9	—	使用できません	
8	ZP	ポジションゾーン出力モニタ	0:範囲外 1:範囲内 各ポジション毎に設定されたゾーンの範囲内に現在位置があるときに 1 となります。範囲外では 0 になります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
7	—	使用できません	
6	—	使用できません	
5	—	使用できません	
4	—	使用できません	
3	—	使用できません	
2	—	使用できません	
1	Z2	ゾーン出力モニタ 2	0:範囲外 1:範囲内 パラメータのゾーン境界 2 で設定された範囲内に現在位置があるときに 1 となります。範囲外では 0 になります。 本出力は原点復帰完了後に有効となります。 サーボオフ中も有効です。
0	Z1	ゾーン出力モニタ 1	0:範囲外 1:範囲内 パラメータのゾーン境界 1 で設定された範囲内に現在位置があるときに 1 となります。範囲外では 0 になります。 本出力は原点復帰完了後に有効となります。 サーボオフ中も有効です。

(18) ポジション番号ステータスレジスタ内容(アドレス = 9014_H)(POSS)

ビット	記号	名 称	機 能
15	—	使用できません	
14	—	使用できません	
13	—	使用できません	
12	—	使用できません	
11	—	使用できません	
10	—	使用できません	
9	PM512	完了ポジション番号ステータスビット 512	<p>位置決めが完了したポジション番号を示します。 (PIO パターン 4、5 電磁弁モード以外の時に有効)</p> <p>完了ポジションは、バイナリコードで読み出されます。 目標位置の周囲(正方向も負方向も位置決め幅分の範囲)に現在位置が入ると完了ポジション番号を読み出すことができます。それ以外では、全て 0 が読み出されます。</p> <p>サーボ OFF では全て 0 となりますが、再びサーボ ON した時に目標位置の周囲内に現在位置が入っていれば、完了ポジション番号も再び有効となります。</p> <p>押付け時は、押付け完了 および、押付け空振りのどちらでも完了ポジションを読み出すことができます。</p>
8	PM256	完了ポジション番号ステータスビット 256	
7	PM128	完了ポジション番号ステータスビット 128	
6	PM64	完了ポジション番号ステータスビット 64	
5	PM32	完了ポジション番号ステータスビット 32	
4	PM16	完了ポジション番号ステータスビット 16	
3	PM8	完了ポジション番号ステータスビット 8	
2	PM4	完了ポジション番号ステータスビット 4	
1	PM2	完了ポジション番号ステータスビット 2	
0	PM1	完了ポジション番号ステータスビット 1	

(19) 拡張システムステータスレジスタの内容(アドレス = 9015_H)(SSSE)

ビット	記号	名 称	機 能
15	—	使用できません	
14	—	使用できません	
13	—	使用できません	
12	—	使用できません	
11	ALMC	コールドスタートレベルアラーム	0:通常 1:コールドスタートレベルアラーム発生中 コールドスタートレベルのアラーム発生で1となります。運転再開には、アラーム要因を解除し、電源再投入が必要です。
10	—	使用できません	
9	—	使用できません	
8	RTC	RTC(カレンダー)機能使用	0:RTC(カレンダー)機能未使用 1: RTC(カレンダー)機能使用 ※対応機種:ERC3、PCON-CA/CFA
7	—	使用できません	
6	—	使用できません	
5	—	使用できません	
4	—	使用できません	
3	—	使用できません	
2	—	使用できません	
1	—	使用できません	
0	—	使用できません	

通信

4.3.3 Modbus ステータスの構造

Modbus ステータスは次のように配置されています。

0000 _H	(システム用) ^(注)
0100 _H ? 010F _H	デバイスステータスレジスタ 1 [DSS1]
0110 _H ? 011F _H	デバイスステータスレジスタ 2 [DSS2]
0120 _H ? 012F _H	拡張デバイスステータスレジスタ [DSSE]
0130 _H ? 013F _H	ポジション番号ステータスレジスタ [POSS]
0140 _H ? 014F _H	ゾーンステータスレジスタ [ZONS]
0150 _H ? 015F _H	入力ポートモニタレジスタ [DIPM]
0160 _H ? 016F _H	出力ポートモニタレジスタ [DOPM]
0170 _H ? 017F _H	特殊入力ポートモニタレジスタ [SIPM]
0180 _H ? 018F _H	拡張システムステータスレジスタ [SSSE]
	(システム用) ^(注)
0400 _H ? 040F _H	デバイス制御レジスタ 1 [DRG1]
0410 _H ? 041F _H	デバイス制御レジスタ 2 [DRG2]
0420 _H ? 042F _H	拡張デバイス制御レジスタ [DRGE]
0430 _H ? 043F _H	ポジション番号指定レジスタ [POSR]
FFFF _H	(システム用) ^(注)

注 システム用領域は通信に使用できません。

4.3.4 Modbus ステータス詳細

アドレス[HEX]	エリア名称	内容	記号	掲載頁					
				RTU		ASCII			
0000~00FF	システム用								
0100	デバイス	EMG ステータス	EMGS	(92)	40	(224)	40		
0101	ステータス	セーフティ速度有効ステータス	SFTY						
0102	レジスタ 1	コントローラレディステータス	PWR						
0103	(DSS1)	サーボ ON ステータス	SV						
0104		押付け空振り	PSFL						
0105		重故障ステータス	ALMH						
0106		軽故障ステータス	ALML						
0107		アブソエラーステータス	ABER						
0108		ブレーキ強制解除ステータス	BKRL						
0109		使用できません							
010A		一時停止ステータス	STP					40	40
010B		原点復帰ステータス	HEND						
010C		位置決め完了ステータス	PEND						
010D		ロードセルキャリブレーション完了	CEND						
010E		ロードセルキャリブレーションステータス	CLBS						
010F		使用できません							
0110	デバイス	使用できません		(94)		(226)			
0111	ステータス	使用できません							
0112	レジスタ 2	負荷出力判定ステータス	LOAD					41	41
0113	(DSS2)	トルクレベルステータス	TRQS						
0114		ティーチモードステータス	MODS						
0115		ポジションデータ取込み指令ステータス	TEAC						
0116		ジョグ+ステータス	JOG+						
0117		ジョグ-ステータス	JOG-						
0118		完了ポジション 7	PE7						
0119		完了ポジション 6	PE6						
011A		完了ポジション 5	PE5						
011B		完了ポジション 4	PE4						
011C		完了ポジション 3	PE3						
011D		完了ポジション 2	PE2						
011E		完了ポジション 1	PE1						
011F		完了ポジション 0	PE0						
0120	拡張	非常停止ステータス	EMGP	(96)	42	(228)	42		
0121	デバイス	モータ電圧低下ステータス	MPUV						
0122	ステータス	運転モードステータス	RMDS						
0123	レジスタ	使用できません							
0124	(DSSE)	原点復帰中ステータス	GHMS					42	42
0125		押付け動作中	PUSH						
0126		励磁検出ステータス	PSNS						
0127		PIO/Modbus 切替ステータス	PMSS						
0128		使用できません							
0129		使用できません							
012A		移動中信号	MOVE					42	42
012B~012F		使用できません							

通信

アドレス[HEX]	エリア名称	内容	記号	掲載頁				
				RTU		ASCII		
0130~0135	ポジション 番号 ステータス レジスタ (POSS)	使用できません		(112)		(244)		
0136		完了ポジション番号ステータスビット 512	PM512		46		46	
0137		完了ポジション番号ステータスビット 256	PM256					
0138		完了ポジション番号ステータスビット 128	PM128					
0139		完了ポジション番号ステータスビット 64	PM64					
013A		完了ポジション番号ステータスビット 32	PM32					
013B		完了ポジション番号ステータスビット 16	PM16					
013C		完了ポジション番号ステータスビット 8	PM8					
013D		完了ポジション番号ステータスビット 4	PM4					
013E		完了ポジション番号ステータスビット 2	PM2					
013F		完了ポジション番号ステータスビット 1	PM1					
0140		ゾーン ステータス レジスタ (ZONS)	使用できません		(110)		(242)	
0141			リミットセンサ出力モータ 2	LS2		45		45
0142	リミットセンサ出力モータ 1		LS1					
0143	リミットセンサ出力モータ 0		LS0					
0144~0146	使用できません							
0147	ポジションゾーン出力モータ		ZP		45		45	
0148~014D	使用できません							
014E	ゾーン出力モータ 2		Z2		45		45	
014F	ゾーン出力モータ 1		Z1					
0150 ~ 015F	入力ポート モニタレジスタ (DIPM)	PIO コネクタピン番号 20A(IN15)~ PIO コネクタピン番号 5A(IN0)			84		216	
0160~ 016F	出力ポート モニタレジスタ (DOPM)	PIO コネクタピン番号 16B(OUT15)~ PIO コネクタピン番号 1B(OUT0)			88		220	
0170	特殊入力 ポートモータ レジスタ (SIPM)	使用できません		(108)		(240)		
0171		指令ハルス NP 信号状態	NP		44		44	
0172		使用できません						
0173		指令ハルス PP 信号状態	PP		44		44	
0174~0176		使用できません						
0177		モードスイッチ状態	MDSW		44		44	
0178~017A		使用できません						
017B		ハルト切断センサモータ	BLCT		44		44	
017C		原点確認センサモータ	HMCK		44		44	
017D		オーバトラベルセンサモータ	OT					
017E		クリーブセンサモータ	CREP					
017F		リミットセンサモータ	LS					
0180~0183	拡張システムス テータスレジスタ (SSSE)	使用できません		(114)		(246)		
0184		コールドスタートレベルアラーム	ALMC		47		47	
0185~0186		使用できません						
0187		RTC 使用(ERC3、PCON-CA/CFA 専用)	RTC		47		47	
0188~018F		使用できません						
0190~03FF	システム用							

アドレス[HEX]	エリア名称	内容	記号	掲載頁								
				RTU		ASCII						
0400	デバイス 制御 レジスタ 1 (DRG1)	EMG 動作指定	EMG	(153)	33	(285)	33					
0401		セーフティ速度指令	SFTY									
0402		使用できません										
0403		サーボ ON 指令	SON									
0404~0406		使用できません										
0407		アラームリセット指令	ALRS									
0408		ブレーキ強制解除指令	BKRL									
0409		使用できません										
040A		一時停止指令	STP									
040B		原点復帰指令	HOME									
040C		位置決め動作起動指令	CSTR									
040D~040F		使用できません										
0410		デバイス 制御 レジスタ 2 (DRG2)	使用できません						(153)	34	(285)	34
0411			ジョグ/インテック切替					JISL				
0412~0413	使用できません											
0414	ティーチモード指令		MOD									
0415	ポジションデータ取込み指令		TEAC									
0416	ジョグ+指令		JOG+									
0417	ジョグ指令		JOG-									
0418	スタートポジション 7		ST7									
0419	スタートポジション 6		ST6									
041A	スタートポジション 5		ST5									
041B	スタートポジション 4		ST4									
041C	スタートポジション 3		ST3									
041D	スタートポジション 2		ST2									
041E	スタートポジション 1		ST1									
041F	スタートポジション 0		ST0									
0420~0425	拡張		使用できません									
0426	デバイス	ロードセルキャリブレーション指令	CLBR	146					278			
0427	制御	PIO/Modbus 切換え指定	PMSL	148					280			
0428~042B	レジスタ	使用できません		150					282			
042C	(DRGE)	減速停止	STOP									
042D~042F	使用できません											
0430~0435	ポジション 番号指定 レジスタ (POSR)	使用できません		(153)	35	(285)	35					
0436		ポジション指令ビット 512	PC512									
0437		ポジション指令ビット 256	PC256									
0438		ポジション指令ビット 128	PC128									
0439		ポジション指令ビット 64	PC64									
043A		ポジション指令ビット 32	PC32									
043B		ポジション指令ビット 16	PC16									
043C		ポジション指令ビット 8	PC8									
043D		ポジション指令ビット 4	PC4									
043E		ポジション指令ビット 2	PC2									
043F	ポジション指令ビット 1	PC1										
0440~FFFF	システム用											

通信

5 Modbus RTU



RTU

5.1 メッセージフレーム(クエリ、レスポンス)

スタート	アドレス	ファンクションコード	データ	CRC チェック	エンド
サイレント インターバル	1 byte	1 byte	n byte	2 byte	サイレント インターバル

(1) スタート

3.5 文字(キャラクタ)以上のサイレント-インターバル(無通信時間)です。

(1 文字(キャラクタ) = 10 bit)

例: 9600bps の時 $(10 \times 3.5) \text{bit} \times 1 / 9600 \text{bps} = 3.65 \text{ms}$

注 レスポンスタイムアウトエラーが発生してしまう場合、パラメータの「No.45 サイレントインターバル倍率」または、「No.17 従局トランスミッタ活性化最小時間」を当社ティーチングツールを使用して適宜変更してください。

(2) アドレス

接続されている RC コントローラのアドレス(01_H~10_H)を指定します。

アドレス = 軸番号 + 1

で設定します。

⚠ 注意 アドレスは軸番号と同じでないため 設定にご注意ください。

(3) ファンクション

RC コントローラで使用可能なファンクションコード、および機能を示します。

コード (Hex)	名称	機能
01H	Read Coil Status	コイル、DO の読出し
02H	Read Input Status	入力ステータス、DI の読出し
03H	Read Holding Registers	保持レジスタの読出し
04H	Read Input Registers	入力レジスタの読出し
05H	Force Single Coil	コイル、DO への 1 点書込み
06H	Preset Single Register	保持レジスタへの書込み
07H	Read Exception Status	例外ステータス読出し
0FH	Force Multiple Coils	複数コイル、DO への一括書込み
10H	Preset Multiple Registers	複数保持レジスタへの一括書込み
11H	Report Slave ID	スレーブ ID 問い合わせ
17H	Read / Write Registers	レジスタへの読出し、書込み

注 本書では マークのファンクションコードを説明しています。

(参考)ROBONET ゲートウェイでは 3 種のファンクションコード(03_H、06_H、10_H)をサポートしていません。[別冊の ROBONET 取扱説明書を参照]

(4) データ

ファンクションコードで指示されたデータを付加する場合に用います。ファンクションコードでデータ付加の指示が無い場合は、データ無しも許されます。

(5) CRC チェック

RTU モードでは、メッセージには CRC 方式に基づいてメッセージ全体の内容がチェックできるようにエラーチェックが自動的^(注)に付加されています。また、チェックはメッセージ中の個別の文字(キャラクタ)のパリティチェック方式と関連せず行われます。

CRC チェックは、16 ビットのバイナリ値で構成されています。CRC 値は、CRC をメッセージに付加する送信側が計算します。受信側は、メッセージ受信中に CRC を再計算して、その計算結果と送られてきた値と比較します。もし、この 2 つの値が一致しなければ、結果はエラーとなります。

(注)パソコンおよび、Modbus に対応していない PLC をホストとして使用する場合、CRC の計算を行う関数を作成する必要があります。

「8.1.CRC チェック計算」に C 言語によるプログラムを掲載しています。

生成多項式: $x^{16} + x^{15} + x^2 + 1$ (CRC-16 方式)

(参考)オムロン製 PLC CJ1 シリーズの Modbus RTU 通信をサポートしている FINS コマンドでは、CRC の計算が自動的に行われます。

(6) エント

3.5 文字(キャラクタ)以上のサイレント・インタバル(無通信時間)です。

(注)レスポンスタイムアウトエラーが発生してしまう場合、パラメータの「No.45 サイレントインターバル倍率 または、No.17 従局トランスミッタ活性化最小時間」を当社ティーチングツールを使用して適宜変更してください。

(7) ブロードキャスト

アドレスを 00_H で指定すると接続されている全ての軸に 同一内容のクエリを送信することができます。この場合 RC コントローラからレスポンスは返信されません。

本機能は使用できるファンクションコード等に制限がありますので、十分注意してご利用ください。使用できるファンクションコードは、「5.2.RTU モードクエリ一覧」でご確認ください。

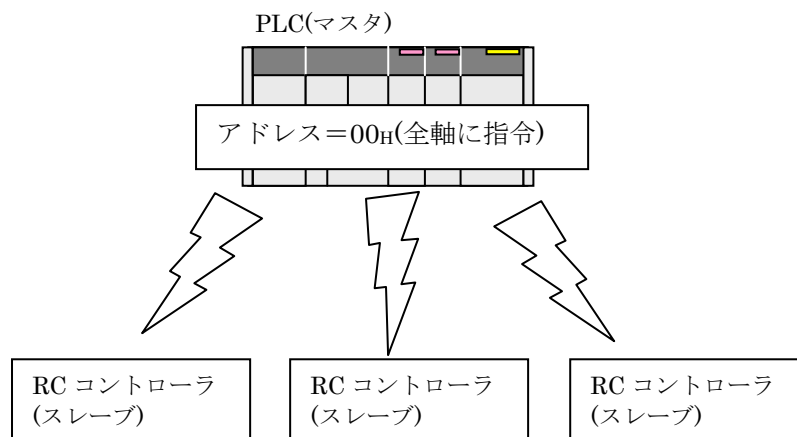


図 5.1

 **注意**

RC コントローラの送受信バッファサイズは、それぞれ 256 バイトとなっています。ホスト側から送信する伝文は受信バッファを、データをリクエストする場合は送信バッファを、それぞれオーバーしないように計算してください。

5.2 RTUモード クエリー一覧

FC:ファンクションコード

PIO:パラレル I/O(I/O コネクタの入出力)

PIOとの併用欄、ブロードキャスト欄の○印は、PIOとの併用、ブロードキャストが有効なクエリを表しています

FC	機能	記号	機能概要	PIOとの併用	ブロードキャスト	頁
03	ファンクションコード 03 複数レジスタ読み込み	無し	ファンクション 03 を使用するレジスタを連続的に複数読出すこと可能です。	○		61
03	アラーム詳細内容 の読み取り	ALA0 ALCO ALTO	最後に発生したアラームコード、アラームアドレス、詳細コード、アラーム発生時刻(経過時間)を読み取ります。	○		65
03	ポジションデータ の読み取り	右記 参照	指定したNo.のポジションデータを読み取ります。 (PCMD,INP,VCMD,ZNMP,ZNLP,ACMD,DCMD,PPOW, LPOW,CTLF)	○		67
03	通算移動回数 の読み取り	TLMC	通算移動回数を読み取ります。	○		70
03	通算走行距離 の読み取り	ODOM	通算走行距離を 1m 単位で読み取ります。	○		72
03	現在時刻の読み取り	TIMN	現在時刻を読み取ります。(PCON-CA/CFA、SCON-CA 専用)	○		74
03	ファン通算駆動時間 の読み取り	TFAN	ファンの通算駆動時間を読み取ります。(PCON-CFA 専用)	○		78
03	現在位置の読み取り	PNOW	アクチュエータの現在位置を 0.01mm 単位で読出します	○		80
03	現在発生アラームコード の読み取り	ALMC	現在発生中のアラームコードを読出します。	○		82
03	I/Oポート入力状態 の読み取り	DIPM	PIO 入力ポート ON/OFF 状態を読出します。	○		84
03	I/Oポート出力状態 の読み取り	DOPM	PIO 出力ポート ON/OFF 状態を読出します。	○		88
03	コントローラ状態信号 の読み取り 1 (デバイスステータス 1) (運転準備ステータス)	DSS1	次の 14 項目の状態(ステータス)を読出します。 ① 非常停止 ② セーフティー速度有効/無効 ③ コントローラレディ ④ サーボ ON/OFF ⑤ 押付け空振り ⑥ 重故障 ⑦ 軽故障 ⑧ アブソエラー ⑨ ブレーキ ⑩ 一時停止 ⑪ 原点復帰完了 ⑫ 位置決め完了 ⑬ ロードセルキャリブレーション完了 ⑭ ロードセルキャリブレーションステータス	○		92

03	コントローラ状態信号 の読み取り2 (デバイスステータス2) (運転情報1ステータス)	DSS2	次の15項目の状態(ステータス)を読出します。 ① イネーブル ② 負荷出力判定(検定範囲負荷電流閾値) ③ トルクレベル(負荷電流閾値) ④ ティーチモード(通常/ティーチ) ⑤ ポジションデータ取込(通常/完了) ⑥ ジョグ+(通常/指令中) ⑦ ジョグ-(通常/指令中) ⑧ 完了ポジション7 ⑨ 完了ポジション6 ⑩ 完了ポジション5 ⑪ 完了ポジション4 ⑫ 完了ポジション3 ⑬ 完了ポジション2 ⑭ 完了ポジション1 ⑮ 完了ポジション0	○		94
03	コントローラ状態信号 の読み取り3 (拡張デバイス ステータス) (運転情報2ステータス)	DSSE	次の9項目の状態(ステータス)を読出します。 ① 非常停止(非常停止入力ポート) ② モータ電圧低下 ③ 運転モード(AUTO/MANU) ④ 原点復帰中 ⑤ 押付け動作中 ⑥ 励磁検出 ⑦ PIO/Modbus 切替 ⑧ ポジションデータ書込み完了ステータス ⑨ 移動中	○		96
03	コントローラ状態信号 の読み取り4 (システムステータス) (コントローラステータス)	STAT	次の7項目の状態(ステータス)を読出します。 ① 自動サーボ OFF 中 ② 不揮発メモリアクセス中 ③ 運転モード(AUTO/MANU) ④ 原点復帰完了 ⑤ サーボ ON/OFF ⑥ サーボ指令 ⑦ 駆動源 ON(通常/しゃ断中)	○		98
03	現在速度 の読み取り	VNOW	アクチュエータの現在速度を 0.01mm/sec 単位で読出します	○		100
03	電流値の読み取り	CNOW	アクチュエータモータ外ル電流指令値を 1mA 単位で読出します	○		102
03	偏差の読み取り	DEVI	1ms 周期毎の偏差量を 1pulse 単位で読出します	○		104
03	電源投入後の積算 時間の読み取り	STIM	コントローラ電源投入時からの積算時間を 1msec 単位で読出します。	○		106

FC	機能	記号	機能概要	PIOとの併用	プロードキャスト	頁
03	特殊入力ポートの入力信号状態の読み取り (センサ入力ステータス)	SIPM	次の 8 項目の状態(ステータス)を読み出します。 ① 指令パルス NP ② 指令パルス PP ③ モードスイッチ ④ ヘルト切断センサ ⑤ 原点確認センサ ⑥ オーハトラベルセンサ ⑦ クリープセンサ ⑧ リミットセンサ	○		108
03	ゾーン出力信号の読み取り	ZONS	次の 6 項目の状態(ステータス)を読み出します。 ① LS2(PIO パターン電磁弁モード[3 点タイプ]) ② LS1(PIO パターン電磁弁モード[3 点タイプ]) ③ LS0(PIO パターン電磁弁モード[3 点タイプ]) ④ ポジションゾーン ⑤ ゾーン 2 ⑥ ゾーン 1	○		110
03	位置決め完了ポジションNo.の読み取り	POSS	次の 9 項目の状態(ステータス)を読み出します。 ① 完了ポジション番号ビット 256 ② 完了ポジション番号ビット 128 ③ 完了ポジション番号ビット 64 ④ 完了ポジション番号ビット 32 ⑤ 完了ポジション番号ビット 16 ⑥ 完了ポジション番号ビット 8 ⑦ 完了ポジション番号ビット 4 ⑧ 完了ポジション番号ビット 2 ⑨ 完了ポジション番号ビット 1	○		112
03	コントローラ状態信号の読み取り 5	SSSE	次の 2 項目の状態(ステータス)を読み出します。 ① コールドスタートレベルアラームの発生/未発生 ② RTC(カレンダー)機能の使用/未使用 (ERC3、PCON-CA/CFA 限定)	○		114
03	カフィードバックデータの読み取り	FBFC	現在のロードセルの測定値を 0.01N 単位で読み出します。	○		116
05	セーフティ速度有効/無効切替	SFTY	有効/無効モード切替を指令します。		○	119
05	サーボ ON/OFF	SON	サーボ ON/OFF を指令します。		○	122
05	アラームリセット	ALRS	アラームリセット/残移動量キャンセルを指令します。		○	124
05	ブレーキ強制解除	BKRL	ブレーキ強制解除を指令します。		○	126
05	一時停止	STP	一時停止を指令します。		○	128
05	原点復帰	HOME	原点復帰動作を指令します。		○	130
05	位置決め動作起動指令	CSTR	ポジションNo.指定移動時のスタート信号です。		○	132

FC	機能	記号	機能概要	PIOとの併用	プロトキャスト	頁
05	ジヨグ/インテグ切替	JISL	ジヨグ/インテグモード切替を行います。		○	134
05	ティーチモード指令	MOD	通常/教示モード切替を行います。		○	136
05	ポジションデータ取込み指令	TEAC	教示モード時、現在位置取込指令を行います		○	138
05	ジヨグ+指令	JOG+	反原点方向にジヨグ/インテグ動作を指令します。		○	140
05	ジヨグ-指令	JOG-	原点方向にジヨグ/インテグ動作を指令します。		○	142
05	スタートポジション 0~7 《ST0~ST7》 移動指令	ST0~ ST7	電磁弁モード時に有効なポジションNo.指定 この指令だけでアクチュエータが動作可能です。		○	144
05	ロードセルキャリブレーション 指令	CLBR	ロードセルのキャリブレーションを行います。		○	146
05	PIO/Modbus 切替設定	PMSL	PIO 外部指令信号の有効/無効切替指令		○	148
05	減速停止	STOP	移動中のアクチュエータを減速停止させる事が出来ます。		○	150
06	制御情報の直接書き込み		コントローラのレジスタの内容を変更(書き込み)します。		○	153
10	直値移動指令	無し	目標位置、位置決め幅、速度、加減速度、押付け、制御設定を1つの伝文で送信し、動作させる事が可能です。通常移動、相対移動、押付け動作が可能です。		○	157
10	ポジションテーブルデータ 書き込み	無し	指定された軸、ポジションNo.のデータを全て変更することが可能です。		○	175
不定	例外レスポンス	無し	伝文内容が不正データであった場合のレスポンスです。			316

5.3 データ、ステータス読み取り(使用ファンクションコード 03)

5.3.1 レジスタの連続複数読み取り

(1) 機能

スレーブのレジスタの内容を読み取ります。

ブロードキャストはサポートされていません。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、RTUモード使用時256バイト中5バイト(スレーブアドレス+ファンクションコード+データバイト数+エラーチェック)を除く251バイト分最大125レジスタ分(1レジスタ2バイト使用)のデータ照会が可能です。連続したアドレスの複数レジスタを一度の送受信で照会することも可能です。

アドレス[H]	記号	名称	符号	レジスタサイズ	バイト
0500	ALAO	アラーム詳細コード		1	2
0501	ALAO	アラームアドレス		1	2
0502	—	常に0	—	1	2
0503	ALCO	アラームコード		1	2
0504,0505	ALTO	アラーム発生時刻		2	4
1000 ~ 3FFF	PCMD	目標位置	○	2	4
	INP	位置決め幅	○	2	4
	VCMD	速度指令		2	4
(注)小さいポ ジションNo.から 順次割付けさ れています。	ZNMP	個別ゾーン境界+側	○	2	4
	ZNLP	個別ゾーン境界-側	○	2	4
	ACMD	加速度指令		1	2
	DCMD	減速度指令		1	2
	PPOW	押付け時電流制限値		1	2
	LPOW	負荷電流閾値		1	2
	CTLF	制御フラグ指定		1	2
8400,8401	TLMC	通算移動回数 ^(注1)		2	4
8402,8403	ODOM	通算走行距離 ^(注1)		2	4
841A,841B	TIMN	現在時刻(SCON-CA 専用)		2	4
8420,8421	TIMN	現在時刻(PCON-CA/CFA 専用)		2	4
842E,842F	TFAN	FAN 通算駆動時間(PCON-CFA 専用)		2	4
9000,9001	PNOW	現在位置モニタ	○	2	4
9002	ALMC	現在発生アラームコード照会		1	2
9003	DIPM	入力ポート照会		1	2
9004	DOPM	出力ポートモニタ照会		1	2
9005	DSS1	デバイスステータス1照会		1	2
9006	DSS2	デバイスステータス2照会		1	2
9007	DSSE	拡張デバイスステータス照会		1	2
9008,9009	STAT	システムステータス照会		2	4

アドレス[H]	記号	名称	符号	レジスタサイズ	バイト
900A,900B	VNOW	現在速度モニタ	○	2	4
900C,900D	CNOW	電流値モニタ	○	2	4
900E,900F	DEVI	偏差モニタ	○	2	4
9010,9011	STIM	システムタイマ照会		2	4
9012	SIPM	特殊入力ポート照会		1	2
9013	ZONS	ゾーンステータス照会		1	2
9014	POSS	位置決め完了ポジションNo.ステータス照会		1	2
9015	SSSE	拡張システムステータスレジスタ		1	2
901E	FBFC	カフィードバックデータモニタ	○	2	4

注1 PCON-CA/CFA、SCON-CA、ERC3専用

(3) クエリフォーマット

クエリメッセージでは、読み取りを開始するレジスタのアドレスとレジスタのバイト数を指定します。

1レジスタ(1アドレス)=2バイト=16ビットデータ

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート		なし	サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタ読出しコード
開始アドレス[H]	2	任意	5.3.1(2)開始アドレス 一覧参照
レジスタの数[H]	2	任意	開始アドレス一覧参照
エラーチェック[H]	2	CRC(16ビット)	
エンド		なし	サイレントインターバル
合計バイト数	8		

RTU
ファンクション
コード 03

(4) レスポンスフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート			サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタ読出しコード
データバイト数[H]	1		クエリ指定レジスタの バイト数の合計
データ1[H]	クエリ指定レジスタのバ イト数		
データ2[H]	クエリ指定レジスタのバ イト数		
データ3[H]	クエリ指定レジスタのバ イト数		
データ4[H]	クエリ指定レジスタのバ イト数		
:	:		
:	:		
エラーチェック[H]	2	CRC(16ビット)	サイレントインターバル
エンド		なし	
合計バイト数	最大256		

(5) 使用例

軸No.0コントローラのアドレス9000_H～9009_Hまでを照会した使用例を示します。

●クエリ(前後にサイレントインターバルが入ります)

01 03 90 00 00 0A E8 CD

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9000
レジスタの数[H]	000A (10レジスタ)
エラーチェック[H]	E8CD (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 14 00 00 00 00 00 00 00 00 00 00 00 00 00 6E 00 60 18 80 00 23 C7 00 00 00 19 18 A6

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	14 (20バイト= 10レジスタ)
データ1[H]	00 00 00 00 (現在位置モータ)
データ2[H]	00 00 (現在発生アラームコード照会)
データ3[H]	00 00 (入力ポート照会)
データ4[H]	6E 00 (出力ポートモータ照会)
データ5[H]	60 18 (デバイスステータス1照会)
データ6[H]	80 00 (デバイスステータス2照会)
データ7[H]	23 C7 (拡張デバイスステータス照会)
データ8[H]	00 00 00 19 (システムステータス照会)
エラーチェック[H]	18A6 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.2 アラーム詳細内容の読み取り《ALA0、ALC0、ALT0》

(1) 機能

最後に発生したアラームコード、アラーム詳細コードおよびアラーム発生時刻を読み取ります。アラームが発生していない場合は0_Hです。[詳細は、4.3.2(1)～(3)項を参照]

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ～10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	0500	アラーム詳細コード
レジスタの数[H]	2	0006	アドレス0500 _H ～0505 _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

RTU
ファンクション
コード 03

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ～10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	0C	6レジスタ呼出=12バイト
データ1[H]	4	アラーム詳細コード アラームアドレス	アラーム詳細コード(0500 _H) [Hex] アラームアドレス(0501 _H) [Hex]
データ2[H]	4	アラームコード	アラームコード[Hex]
データ3[H]	4	アラーム発生時刻 ^(注1)	アラーム発生時刻[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	17		

注1 RTC(カレンダー機能)搭載機種でRTC有効の場合①と、RTC無効、またはRTCが無い機種②ではデータの内容が異なります。

①アラーム発生時刻を示します。 ②電源投入後からの経過時間[msec]を示します。

(4) 使用例

軸No.0コントローラで最後に発生したアラーム内容(アドレス0500_H~0505_H)を読み取りした使用例を示します

●クエリ(前後にサイレントインターバルが入ります)

01 03 05 00 00 06 C5 04

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	0500
レジスタの数[H]	0006
エラーチェック[H]	C504 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 0C 00 00 FF FF 00 00 00 E8 17 2C 64 3F 2D CD

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	0C(12バイト= 6レジスタ)
データ1[H]	00 00(アラーム詳細コード)
データ2[H]	FF FF(アラームアドレス)
データ3[H]	00 00 00 E8(アラームコード)
データ4[H]	17 2C 64 3F(アラーム発生時刻)
エラーチェック[H]	2DCD (CRC計算による)
エンド	サイレントインターバル

アラーム詳細コード:0000_H・・詳細コード無し

アラームアドレス:FFFF_H・・無効(詳細コード無し)

アラームコード:00E8_H=0E8・・エンコード AB 相断線エラー

アラーム発生時刻:172C643F_H(変換)⇒2012/04/26 19:53:35[変換は、4.3.2(3)項参照]

注1 レスポンス例のデータは一例であり、各種状態により異なります。

注2 アラームコードの詳細内容は、各コントローラの取扱説明書を参照ください。

5.3.3 ポジションデータの読み取り 《PCMD、INP、VCMD、ZNMP、ZNLP、ACMD、DCMD、PPOW、LPOW、CTLF》

(1) 機能

指定したポジションNo.に設定された値を読み取ります。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、RTUモード使用時256バイト中5バイト(スレーブアドレス+ファンクションコード+データバイト数+エラーチェック)を除く251バイト分最大125レジスタ分(1レジスタ2バイト使用)のデータ照会が可能です。連続したアドレスの複数レジスタを一度の送受信で照会することも可能です。

アドレス [H]	各ポジションNo. の先頭アドレス [H]	先頭アド レスから のオフセット [H]	記号	レジスタ名称	符号	レジスタ サイズ*	バイト	単位
1000 ~ 3FFF	先頭アドレス= 1000 _H +(16 ×ポジションNo.)	+0	PCMD	目標位置	○	2	4	0.01mm
		+2	INP	位置決め幅	○	2	4	0.01mm
		+4	VCMD	速度指令		2	4	0.01mm/s
		+6	ZNMP	個別ゾーン境界 +側	○	2	4	0.01mm
		+8	ZNLP	個別ゾーン境界 -側	○	2	4	0.01mm
		+A	ACMD	加速度指令		1	2	0.01G
		+B	DCMD	減速度指令		1	2	0.01G
		+C	PPOW	押付け時電流 制限値		1	2	% (100%=FF _H)
		+D	LPOW	負荷電流閾値		1	2	% (100%=FF _H)
		+E	CTLF	制御フラグ指定		1	2	

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_{\text{H}} + (16 \times \text{ポジション No.})_{\text{H}} + \text{アドレス(オフセット値)}_{\text{H}}$$

例 ポジションNo.200 の速度指令レジスタを変更したい場合

$$1000_{\text{H}} + (16 \times 200 = 3200)_{\text{H}} + 4_{\text{H}}$$

$$= 1000_{\text{H}} + \text{C80}_{\text{H}} + 4_{\text{H}}$$

$$= 1\text{C84}_{\text{H}}$$

『1C84』がクエリ開始アドレス部入力値になります。

注 最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

(3) クエリフォーマット

クエリメッセージでは、読み取りを開始するレジスタのアドレスとレジスタのバイト数を指定します。

1レジスタ(1アドレス)=2バイト=16ビットデータ

フィールド名称	RTUモード 8ビットデータ	データ数 (バイト数)	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	任意	1	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	03	1	レジスタの読出し
開始アドレス[H]	任意	2	(2)開始アドレス一覧参照
レジスタの数[H]	任意	2	
エラーチェック[H]	CRC(16ビット)	2	
エンド	なし		サイレントインターバル
合計バイト数		8	

(4) レスポンスフォーマット

フィールド名称	RTUモード 8ビットデータ	データ数 (バイト数)	備考
スタート			サイレントインターバル
スレーブアドレス[H]	任意	1	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	03	1	レジスタの読出し
データバイト数[H]		1	クエリ指定レジスタの バイト数の合計
データ1[H]		クエリ指定レジスタの バイト数	
データ2[H]		クエリ指定レジスタの バイト数	
データ3[H]		クエリ指定レジスタの バイト数	
データ4[H]		クエリ指定レジスタの バイト数	
⋮		⋮	
⋮		⋮	
エラーチェック[H]	CRC(16ビット)	2	
エンド	なし		サイレントインターバル
合計バイト数		最大256	

(5) 使用例

軸No.0コントローラのポジションNo.1(アドレス1010_H~1015_H)の目標位置、位置決め幅および速度指令を照会した使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 10 10 00 06 C0 CD

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	1010
レジスタの数[H]	0006 (6レジスタ)
エラーチェック[H]	C0CD(CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 0C 00 00 07 D0 00 00 1F 40 00 00 3A 98 AF C5

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	0C (12バイト= 6レジスタ)
データ1[H]	00 00 07 D0 (目標位置照会)
データ2[H]	00 00 1F 40 (位置決め幅照会)
データ3[H]	00 00 3A 98 (速度指令照会)
エラーチェック[H]	AF C5 (CRC計算による)
エンド	サイレントインターバル

目標位置は、7D0_H→10進数に変換→2000×[単位0.01mm]= 20.00[mm]

位置決め幅は、1F40_H→10進数に変換→8000×[単位0.01mm]=80.00[mm]

速度指令は、3A98_H→10進数に変換→15000×[単位0.01mm]=150.00[mm]

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.4 通算移動回数の読み取り《TLMC》

(1) 機能

通算移動回数を読み取ります。[詳細は、4.3.2(8)項を参照]

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	8400	通算移動回数
レジスタの数[H]	2	0002	アドレス8400 _H ~8401 _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ1[H]	2	通算移動回数	通算移動回数[Hex](上位)
データ2[H]	2	通算移動回数	通算移動回数[Hex](下位)
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの通算移動回数(アドレス8400_H~8401_H)を読み取りした使用例を示します。

●クエリ(前後にサイレントインターバルが入ります)

01 03 84 00 00 02 EC FB

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	8400
レジスタの数[H]	0002
エラーチェック[H]	ECFB (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 02 1F BA 9B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00
データ2[H]	02 1F
エラーチェック[H]	BA9B (CRC計算による)
エンド	サイレントインターバル

通算移動回数は、21F_H→10進数に変換→543[回]

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.5 通算走行距離の読み取り《ODOM》(1m 単位)

(1) 機能

通算走行距離を 1m 単位で読み取ります。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	8402	通算走行距離
レジスタの数[H]	2	0002	アドレス8402 _H ~8403 _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ1[H]	2	通算走行距離	通算走行距離[Hex](上位)
データ2[H]	2	通算走行距離	通算走行距離[Hex](下位)
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの通算走行距離(アドレス8402_H~8403_H)を読み取りした使用例を示します

●クエリ(前後にサイレントインターバルが入ります)

01 03 84 02 00 02 4D 3B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	8402
レジスタの数[H]	0002
エラーチェック[H]	4D3B (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 40 9E 4A 5B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00
データ2[H]	40 9E
エラーチェック[H]	4A5B (CRC計算による)
エンド	サイレントインターバル

通算走行距離は 0000409E_H→10進数に変換→16542m

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.6 現在時刻の読み取り《TIMN》

(1) 機能

現在時刻を読み取ります。

[PCON-CA/CFA および SCON-CA 専用]

(2) ケリフォーマット

フィールド名	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	備考参照	841A:SCON-CA 8420:PCON-CA/CFA
レジスタの数[H]	2	0002	アドレス8402 _H ~8403 _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	現在時刻	時刻への変換は(4)参照
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 読み取ったデータを時刻に変換

読み取ったデータは、コントローラの設定により、現在の時刻または時間を出力します。

- ①RTC(カレンダー機能)を有効に設定している場合、現在時刻になります。
- ②RTCを無効に設定している場合、コントローラ電源投入を基準とした経過時間[sec]になります。

①現在時刻の計算方法

読み取った現在時刻のデータは、基準時刻(2000年1月1日00時00分00秒)からの経過秒を示しています。

基準時刻からの経過秒をS、経過分をM、経過時をH、経過日をD、経過年をYとし、次の式で計算を行います。

$$\begin{aligned} S &= \text{読み込んだ現在時刻のデータ} \\ M &= S/60(\text{小数点以下切捨て}) \\ H &= M/60(\text{小数点以下切捨て}) \\ D &= H/24(\text{小数点以下切捨て}) \\ Y &= D/365.25(\text{小数点以下切捨て}) \\ L(\text{閏年計算}) &= Y/4(\text{小数点切り上げ}) \end{aligned}$$

時刻の秒をSA、分をMA、時をHA、今年になってからの経過日をDA、年をYAとすると、以下の式で時刻を計算できます。

$$\begin{aligned} SA &= S/60 \text{の余り} \\ MA &= M/60 \text{の余り} \\ HA &= H/24 \text{の余り} \\ DA &= D - (Y \times 365 + L) \cdots DA \text{ から一月ごとの日数を減算することで月日を求めます。} \\ YA &= Y + 2000 \text{ (西暦)} \end{aligned}$$

例) 現在時刻のデータが172C1B8B_Hだった場合

【手順1】10進数に変換: S=172C1B8B_H⇒388766603

【手順2】M、H、D、Y、Lを計算します。

$$\begin{aligned} M &= 388766603/60 = 6479443 \\ H &= 6479443/60 = 107990 \\ D &= 107990/24 = 4499 \\ Y &= 4499/365.25 = 12 \\ L &= 12/4 = 3 \end{aligned}$$

【手順3】SA、MA、HA、およびDAを求めます。

$$\begin{aligned} SA &= 388766603/60 \text{の余り} = 23 \\ MA &= 6479443/60 \text{の余り} = 43 \\ HA &= 107990/24 \text{の余り} = 14 \\ DA &= 4499 - (12 \times 365 + 3) \\ &= 116(\text{今年になって116日経過し、現在は117日目}) \\ \text{月日} &= 117 - \{31(1月) - 29(2月) - 31(3月)\} \\ &= 26(4月分を減算すると負数になってしまうので、発生時は4月26日) \\ YA &= 12 + 2000 = 2012 \end{aligned}$$

以上より、現在時刻は、2012年4月26日14時43分23秒となります。

②経過時間の計算方法

例) 現在時刻のデータが E1B8B_H だった場合

10 進数に変換: E1B8B_H ⇒ 924555

したがって、電源投入後 924555sec(256 時間 49 分 15 秒)経過となります。

(5) 使用例

軸No.0のPCON-CAの現在時刻(アドレス8420_H~8421_H)を読み取りした使用例を示します。

●クエリ(前後にサイレントインターバルが入ります)

01 03 84 20 00 02 ED 31

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	8420
レジスタの数[H]	0002
エラーチェック[H]	ED31 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 17 2C 1B 8B 74 D9

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ[H]	17 2C 1B 8B
エラーチェック[H]	74D9 (CRC計算による)
エンド	サイレントインターバル

現在時刻は、2012 年 4 月 26 日 14 時 43 分 23 秒

注 レスポンス例のデータは一例であり、各種状態により異なります。



RTU
ファンクション
コード 03

5.3.7 ファン通算駆動時間の読み取り《TFAN》

(1) 機能

ファンの通算駆動時間を読み取ります。(1 秒単位)

[PCON-CFA 専用]

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	842E	ファン通算駆動時間
レジスタの数[H]	2	0002	アドレス842E _H ~842F _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ1[H]	2	ファン通算駆動時間	ファン通算駆動時間[Hex](上位)
データ2[H]	2	ファン通算駆動時間	ファン通算駆動時間[Hex](下位)
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラのファン通算駆動時間(アドレス842E_H~842F_H)を読み取りした使用例を示します。

●クエリ(前後にサイレントインターバルが入ります)

01 03 84 2E 00 02 8C F2

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	842E
レジスタの数[H]	0002
エラーチェック[H]	8CF2 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 02 AF BB 2F

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00
データ2[H]	02 AF
エラーチェック[H]	BB2F (CRC計算による)
エンド	サイレントインターバル

ファン通算駆動時間は 000002AF_H→10進数に変換→687[秒]

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.8 現在位置の読み取り《PNOW》(0.01mm 単位)

(1) 機能

現在位置を 0.01mm 単位で読み取ります。符号は有効です。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9000	現在位置モニタ
レジスタの数[H]	2	0002	アドレス9000 _H ~9001 _H 呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ1[H]	2	現在位置による	現在位置データ[Hex](上位)
データ2[H]	2	現在位置による	現在位置データ[Hex](下位)
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの現在位置(アドレス9000_H~9001_H)を読み取りした使用例を示します

●クエリ(前後にサイレントインターバルが入ります)

01 03 90 00 00 02 E9 0B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9000
レジスタの数[H]	0002
エラーチェック[H]	E90B (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 0B FE 7C 83

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00
データ2[H]	0B FE
エラーチェック[H]	7C83 (CRC計算による)
エンド	サイレントインターバル

現在位置は『00000BFE_H』→10進数に変換→3070(×0.01mm)→

現在位置は 30.7mm

例 2) 現在位置が『FFFFFFF5_H』と読み取れた時(負の位置)→

FFFFFFF_H-FFFFFFF5_H+1(必ず1を加算)→

10進数に変換→11(×0.01mm)→ 現在位置は-0.11mm

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.9 現在発生アラームコードの読み取り《ALMC》

(1) 機能

コントローラの正常状態、またはアラーム状態(コールドスタートレベル、動作解除レベルおよびメッセージレベル)を示すコードを読み取ります。

正常状態では 00_H が格納されています。

[アラームコードの詳細内容は各コントローラの取扱説明書を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9002	現在発生アラームコード
レジスタの数[H]	2	0001	アドレス9002 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ1[H]	2	アラームコード	アラームコード[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのアラームコード(アドレス9002_H)を読み取りした使用例を示します

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 02 00 01 08 CA

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9002
レジスタの数[H]	0001
エラーチェック[H]	08CA (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 E8 B8 0A

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	00 E8
エラーチェック[H]	B80A (CRC計算による)
エンド	サイレントインターバル

本例で発生している最重要アラームは『0E8』_H・・A,B 相断線検出アラームです。

[アラームコードの詳細内容は各コントローラの取扱説明書を参照]

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.10 I/Oポート入力信号状態読み取り《DIPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート入力値を読み取ります。

RCコントローラが入力として認識しているポートの状態が読み込まれます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9003	入力ポートモニタレジスタ
レジスタの数[H]	2	0001	アドレス9003 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、17アドレス当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ1[H]	2	ポート入力値	ポート入力値[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの入力ポート(アドレス9003_H)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 03 00 01 59 0A

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9003
レジスタの数[H]	0001
エラーチェック[H]	590A (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 90 00 D4 44

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	90 00
エラーチェック[H]	D444 (CRC計算による)
エンド	サイレントインターバル

入力ポートデータ部は『9000』_H →2進数変換『1001000000000000』

↑
↑
 IN15 IN1

注 レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け [詳細は、各 RC コントローラの取扱説明書を参照]

各 RC コントローラの PIO パターンごとのポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)				ERC3(PIOタイプ)		
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン0	PIOパターン1	PIOパターン2
IN0	PC1	ST0	PC1	PC1	PC1	ST0	PC1
IN1	PC2	ST1	PC2	PC2	PC2	ST1	PC2
IN2	PC4	ST2	PC4	PC4	PC4	ST2	PC4
IN3	HOME	0	PC8	PC8	HOME	0	PC8
IN4	CSTR	RES	CSTR	CSTR	CSTR	RES	CSTR
IN5	*STP	*STP	*STP	*STP	*STP	*STP	*STP
IN6	0	0	0	0	0	0	0
IN7	0	0	0	0	0	0	0
IN8	0	0	0	0	0	0	0
IN9	0	0	0	0	0	0	0
IN10	0	0	0	0	0	0	0
IN11	0	0	0	0	0	0	0
IN12	0	0	0	0	0	0	0
IN13	0	0	0	0	0	0	0
IN14	0	0	0	0	0	0	0
IN15	0	0	0	0	0	0	0

ポート	PCON-C/CG/CF/CA/CFA						PCON-CA/CFA	PCON-PL/PO	
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン4	PIOパターン5	PIOパターン6 (パルス列モード)	PIOパターン0	PIOパターン1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	RES	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	TL	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	CSTR	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	DCLR	0	0
IN6	0	MODE	PC64	PC64	ST6	0	BKRL	0	0
IN7	0	JISL	PC128	PC128	0	0	RMOD	0	0
IN8	0	JOG+	0	PC256	0	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0	0

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン0	PIO パターン1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

RTU
ファンクション
コード 03

ポート	SCON-C/CA						SCON-CA		SCON-C/CA
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン6	PIO パターン7	PIOパターン0 (パルス列モード)
IN0	PC1	PC1	PC1	PC1	ST0	ST0	PC1	ST0	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	PC2	ST1	RES
IN2	PC4	PC4	PC4	PC4	ST2	ST2	PC4	ST2	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	PC8	ST3	TL
IN4	PC16	PC16	PC16	PC16	ST4	0	PC16	ST4	CSTR
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0	DCLR
IN6	0	MODE	PC64	PC64	ST6	0	0	0	BKRL
IN7	0	JISL	PC128	PC128	0	0	0	0	RMOD
IN8	0	JOG+	0	PC256	0	0	CLBR	CLBR	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	BKRL	BKRL	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0
IN11	HOME	HOME	HOME	HOME	HOME	0	HOME	HOME	0
IN12	*STP	*STP	*STP	*STP	*STP	0	*STP	*STP	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	CSTR	0	0
IN14	RES	RES	RES	RES	RES	RES	RES	RES	0
IN15	SON	SON	SON	SON	SON	SON	SON	SON	0

5.3.11 I/Oポート出力信号状態の読み取り《DOPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート出力値を読み取ります。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9004	出力ポートモニタレジスタ
レジスタの数[H]	2	0001	アドレス9004 _H のみの呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ1[H]	2	DO出力値	ポート出力値[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0 コントローラの出カポート(アドレス 9004_H)を読み取りした使用例を示します

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 04 00 01 E8 CB

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9004
レジスタの数[H]	0001
エラーチェック[H]	E8CB (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 68 00 97 84

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	68 00
エラーチェック[H]	9784 (CRC計算による)
エンド	サイレントインターバル

入力ポートデータ部は『6800』_H →2進数変換『0110100000000000』

↑
 IN15 IN1

注 レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け [詳細は、各 RC コントローラの取扱説明書を参照]

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)				ERC3(PIOタイプ)		
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン0	PIOパターン1	PIOパターン2
OUT0	PEND	PE0	PEND	PEND	PEND	PE0	PEND
OUT1	HEND	PE1	HEND	HEND	HEND	PE1	HEND
OUT2	ZONE	PE2	ZONE	ZONE	ZONE1	PE2	PZONE/ ZONE1
OUT3	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM
OUT4	0	0	0	0	0	0	0
OUT5	0	0	0	0	0	0	0
OUT6	0	0	0	0	0	0	0
OUT7	0	0	0	0	0	0	0
OUT8	0	0	0	0	0	0	0
OUT9	0	0	0	0	0	0	0
OUT10	0	0	0	0	0	0	0
OUT11	0	0	0	0	0	0	0
OUT12	0	0	0	0	0	0	0
OUT13	0	0	0	0	0	0	0
OUT14	0	0	0	0	0	0	0
OUT15	0	0	0	0	0	0	0

ポート	PCON-C/CG/CF/CA/CFA						PCON-CA/CFA	PCON-PL/PO	
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン4	PIOパターン5	PIOパターン6 (パルス列モード)	PIOパターン0	PIOパターン1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PWR	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	SV	INP	INP/TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	INP	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	HEND	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	TLR	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	*ALM	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	*EMGS	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	RMDS	0	0
OUT8	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1 ^(注)	PZONE/ ZONE1 ^(注)	PM256	PZONE/ ZONE2 ^(注)	PZONE/ ZONE2 ^(注)	ALM1	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	ALM4	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	ALM8	0	0
OUT12	SV	SV	SV	SV	SV	SV	*ALML ^(注)	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	ZONE1	0	0
OUT15	LOAD/ TRQS/ ALML ^(注)	ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	ALML ^(注)	ZONE2	0	0

注 CA/CFA タイプ限定

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン0	PIO パターン1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	TRQS	0	TRQS	TRQS	TRQS	0	0	0

RTU
ファンクション
コード 03

ポート	SCON-C/CA						SCON-CA		SCON-C/CA
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン6	PIO パターン7	PIOパターン0 (パルス列ポート)
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PM1	PE0	PWR
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	PM2	PE1	SV
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	PM4	PE2	INP
OUT3	PM8	PM8	PM8	PM8	PE3	0	PM8	PE3	HEND
OUT4	PM16	PM16	PM16	PM16	PE4	0	PM16	PE4	TLR
OUT5	PM32	PM32	PM32	PM32	PE5	0	TRQS	TRQS	*ALM
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	LOAD	LOAD	*EMGS
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	CEND	CEND	RMDS
OUT8	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1 ^(注)	PZONE/ ZONE1 ^(注)	PM256	PZONE/ ZONE2 ^(注)	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1	PZONE/ ZONE1	ALM1
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	HEND	HEND	ALM4
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	PEND	PEND	ALM8
OUT12	SV	SV	SV	SV	SV	SV	SV	SV	*OVLW ^(注) / *ALML ^(注)
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	ZONE1
OUT15	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	ZONE2

注 CA タイプ限定

5.3.12 コントローラ状態信号の読み取り1《DSS1》

(1) 機能

コントローラのステータスを読み取ります。

[4.3.2 (12) デバイスステータスレジスタ1 内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9005	デバイスステータス レジスタ1
レジスタの数[H]	2	0001	アドレス9005 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1アドレス当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	ステータス1	ステータス1[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9005_H)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 05 00 01 B9 0B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9005
レジスタの数[H]	0001
エラーチェック[H]	B90B (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 70 98 9C 2E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	70 98
エラーチェック[H]	9C2E (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.13 コントローラ状態信号の読み取り2《DSS2》

(1) 機能

コントローラのステータスを読み取ります。

[4.3.2 (13) デバイスステータスレジスタ2 内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9006	デバイスステータス レジスタ2
レジスタの数[H]	2	0001	アドレス9006 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	コントローラ内のステータス
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	ステータス2	ステータス2[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9006_H)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 06 00 01 49 0B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9006
レジスタの数[H]	0001
エラーチェック[H]	490B (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 80 00 D9 84

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	80 00
エラーチェック[H]	D984 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.14 コントローラ状態信号の読み取り3《DSSE》

(1) 機能

コントローラのステータス(拡張デバイス)を読み取ります。

[4.3.2(14) 拡張デバイスステータスレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9007	拡張デバイスステータス レジスタ
レジスタの数[H]	2	0001	アドレス9007 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	拡張ステータス	拡張ステータス[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの拡張デバイスステータス(アドレス9007_H)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 07 00 01 18 CB

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9007
レジスタの数[H]	0001
エラーチェック[H]	18CB (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 33 C7 ED 26

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	33 C7
エラーチェック[H]	ED26 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.15 コントローラ状態信号の読み取り4《STAT》

(1) 機能

コントローラの内部動作状態を読み取ります。

[4.3.2 (15) システムステータスレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9008	システムステータス レジスタ
レジスタの数[H]	2	0002	アドレス9008 _H ~9009 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	コントローラ内のステータス
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	システムステータス	システムステータス[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラのシステムステータス(アドレス9008_H~)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 08 00 02 68 C9

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9008
レジスタの数[H]	0002
エラーチェック[H]	68C9 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 0C 00 17 7A 3E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 0C 00 17
エラーチェック[H]	7A3E (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.16 現在速度の読み取り《VNOW》

(1) 機能

モータ実速度のモニタデータを読み取ります。移動方向により±に変化します。

単位は0.01mm/sです。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	900A	現在速度モニタ
レジスタの数[H]	2	0002	アドレス900A _H ~900B _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	現在速度	現在速度[Hex] 単位は0.01mm/s
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの現在速度モニタ(アドレス900A_H~)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 0A 00 02 C9 09

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	900A
レジスタの数[H]	0002
エラーチェック[H]	C909 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 03 E4 FA 88

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00 03 E4
エラーチェック[H]	FA88 (CRC計算による)
エンド	サイレントインターバル

現在速度は『000003E4』→10進数に変換→996(×0.01mm/s)→
現在速度モニタは 9.96mm/s

例 2)現在速度が『FFFFFF35』と読み取れた時(上の例と反対方向に動作)→
FFFFFFF_H - FFFFFFF35_H + 1(必ず1を加算)→
10進数に変換→203(×0.01mm/s)→
現在速度は 2.03mm/s

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.17 電流値の読み取り《CNOW》

(1) 機能

モータ電流(トルク電流指令値)のモニタデータを読み取ります。

単位は mA です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	900C	電流値モニタ
レジスタの数[H]	2	0002	アドレス900C _H ~900D _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	モータ電流の モニタ	モータ電流のモニタ[Hex] 単位はmA
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの電流値モニタ(アドレス900C_H~)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 0C 00 02 29 08

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	900C
レジスタの数[H]	0002
エラーチェック[H]	2908 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 01 C8 FA 35

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00 01 C8
エラーチェック[H]	FA35 (CRC計算による)
エンド	サイレントインターバル

モータ電流値は『000001C8』→10進数に変換→456

電流モニタ値は 456mA

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.18 偏差の読み取り《DEVI》

(1) 機能

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量を読み取ります。

単位は pulse です。

モータ機械角 1 回転あたりのパルス数は使用エンコーダにより異なります。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	900E	偏差モニタ
レジスタの数[H]	2	0002	アドレス900E _H ~900F _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	偏差モニタ	偏差モニタ[Hex] 単位はpulse
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの偏差モニタ(アドレス900E_H~)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 0E 00 02 88 C8

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	900E
レジスタの数[H]	0002
エラーチェック[H]	88C8 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 00 0B BB F4

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00 00 0B
エラーチェック[H]	BBF4 (CRC計算による)
エンド	サイレントインターバル

偏差モニタは『0000000B』→10進数に変換→11

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量は 11pulse

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.19 電源投入後の積算時間の読み取り《STIM》

(1) 機能

コントローラ電源投入時からの積算時間を読み取ります。

単位は ms です。

ソフトウェアリセットではタイマ値はクリアされません。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレープアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9010	システムタイマ
レジスタの数[H]	2	0002	アドレス9010 _H ~9011 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレープアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	システムタイマ	システムタイマ[Hex] 単位はms
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラのシステムタイマ値(アドレス9010_H～)を読み取りした使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 03 90 10 00 02 E8 CE

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9010
レジスタの数[H]	0002
エラーチェック[H]	E8CE (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 02 7A 72 F8 B6

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 02 7A 72
エラーチェック[H]	F8B6 (CRC計算による)
エンド	サイレントインターバル

システムタイマは『00027A72』→10進数に変換→162418(ms)

コントローラ電源投入時からの積算時間は 162.418 秒

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.20 特殊入力ポートの入力信号状態の読み取り《SIPM》

(1) 機能

通常の入力ポート以外の入力ポートの状態を読み取ります。

[4.3.2(16) 特殊入力ポートモニタレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9012	特殊入力ポートモニタ
レジスタの数[H]	2	0001	アドレス9012 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	特殊ポートモニタ	4.3.2(16) 一覧表参照
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの特異入力ポート(アドレス9012_H)を読み取りした使用例を示します。

クエリ(前後にサイレントインターバルが入ります)

01 03 90 12 00 01 09 0F

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9012
レジスタの数[H]	0001
エラーチェック[H]	090F (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 43 00 89 74

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	43 00
エラーチェック[H]	8974 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.21 ゾーン出力信号の状態読み取り《ZONS》

(1) 機能

ゾーン出力の状態を読み取ります。

[4.3.2 (17) ゾーンステータスレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9013	ゾーンステータス照会
レジスタの数[H]	2	0001	アドレス9013 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	ゾーンステータス	4.3.2(17) 一覧表参照
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのゾーン出力の状態(アドレス9013_H)を読み取りした使用例を示します。

クエリ(前後にサイレントインターバルが入ります)

01 03 90 13 00 01 58 CF

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9013
レジスタの数[H]	0001
エラーチェック[H]	58CF (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 00 B8 44

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	00 00
エラーチェック[H]	B844 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.22 位置決め完了ポジションNo.の読み取り《POSS》

(1) 機能

完了ポジション番号を読み取ります。

[4.3.2(18) ポジション番号ステータスレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9014	ポジション番号ステータス
レジスタの数[H]	2	0001	アドレス9014 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	ポジション番号 ステータス	4.3.2(18) 一覧表参照
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの位置決め完了ポジション(アドレス9014_H)を読み取りした使用例を示します
クエリ(前後にサイレントインターバルが入ります)

01 03 90 14 00 01 E9 0E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9014
レジスタの数[H]	0001
エラーチェック[H]	E90E (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 00 B8 44

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	00 00
エラーチェック[H]	B844 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.23 コントローラ状態信号の読み取り5《SSSE》

(2) 機能

コントローラの内部動作状態を読み取ります。

[4.3.2(19) 拡張システムステータスレジスタ内容を参照]

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	9015	拡張システムステータス レジスタ
レジスタの数[H]	2	0001	アドレス9015 _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	コントローラ内のステータス
データバイト数[H]	1	02	1レジスタ呼出=2バイト
データ[H]	2	拡張システムステータス	拡張システムステータス[Hex]
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの拡張システムステータス(アドレス9015_H)を読み取りした使用例を示します。

- クエリ(前後にサイレントインターバルが入ります)

01 03 90 15 00 01 B8 CE

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	9015
レジスタの数[H]	0001
エラーチェック[H]	B8CE (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンスは以下ようになります。

- レスポンス(前後にサイレントインターバルが入ります)

01 03 02 01 00 B9 D4

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	02(2バイト= 1レジスタ)
データ1[H]	01 00
エラーチェック[H]	B9D4 (CRC計算による)
エンド	サイレントインターバル

注 レスポンス例のデータは一例であり、各種状態により異なります。

5.3.24 カフィードバックデータの読み取り《FBFC》-SCON-CA 専用

(1) 機能

ロードセル測定値(押付け力)のモニタデータを読み取ります。

単位は0.01Nです。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
開始アドレス[H]	2	901E	カフィードバックデータモニタ
レジスタの数[H]	2	0002	アドレス901E _H ~901F _H の呼出
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	1	03	レジスタの読み出し
データバイト数[H]	1	04	2レジスタ呼出=4バイト
データ[H]	4	ロードセル測定値	現在の押付け力[N] 単位は0.01N
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラに接続されているロードセルの現在測定値を読み取りした使用例を示します。

●クエリ(前後にサイレントインターバルが入ります)

01 03 90 0A 00 02 89 0D

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
開始アドレス[H]	901E
レジスタの数[H]	0002
エラーチェック[H]	890D (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 03

レスポンス^(注1)は以下のようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 03 E4 FA 88

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	03
データバイト数[H]	04(4バイト= 2レジスタ)
データ1[H]	00 00 03 E4
エラーチェック[H]	FA88 (CRC計算による)
エンド	サイレントインターバル

例1) 現在のロードセル測定値は『000003E4』→10進数に変換→996(×0.01N)→
現在の押付け力は9.96N

例2) 現在のロードセル測定値が『FFFFFF35』と読み取れた時、(引張り状態^(注2))→
FFFFFFFF_H - FFFFFFF35_H + 1(必ず1を加算)→
10進数に変換→203(×0.01N)→
現在の引張り力^(注2)は2.03N

注1 レスポンス例のデータは一例であり、各種状態により異なります。

注2 ロードセルは、引張り動作には使用できません。

5.4 動作指令 及び、データ書き換え(使用ファンクションコード 05)

5.4.1 コイルへの書込み

(1) 機能

スレーブのDO(Discrete Output)の状態をON/OFFのいずれかに変更(書込み)します。
ブロードキャストの場合には、全スレーブの同じアドレスのコイルを書換えます。

(2) 開始アドレス一覧

アドレス[H]	記号	機能名称
0401	SFTY	セーフティ速度指令
0403	SON	サーボ ON 指令
0407	ALRS	アラームリセット指令
0408	BKRL	ブレーキ強制解除指令
040A	STP	一時停止指令
040B	HOME	原点復帰指令
040C	CSTR	位置決め動作起動指令
0411	JISL	ジョグ/インテグ切替え
0414	MOD	ティーチモード指令
0415	TEAC	ポジションデータ取込み指令
0416	JOG+	ジョグ+指令
0417	JOG-	ジョグ-指令
0418	ST7	スタートポジション 7(電磁弁モード)
0419	ST6	スタートポジション 6(電磁弁モード)
041A	ST5	スタートポジション 5(電磁弁モード)
041B	ST4	スタートポジション 4(電磁弁モード)
041C	ST3	スタートポジション 3(電磁弁モード)
041D	ST2	スタートポジション 2(電磁弁モード)
041E	ST1	スタートポジション 1(電磁弁モード)
041F	ST0	スタートポジション 0(電磁弁モード)
0426	CLBR	ロードセルキャリブレーション指令
0427	PMSL	PIO/Modbus 切替え指定
042C	STOP	減速停止

5.4.2 セーフティ速度有効/無効切替《SFTY》

(1) 機能

ユーザパラメータNo.35の「セーフティ速度」で指定された速度の有効/無効切替を行います。
MANUモード時に有効にしますと、全ての移動指令速度が制限されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0401	セーフティ速度指令
変更データ[H]	2	任意	セーフティ速度有効: FF00 _H セーフティ速度無効: 0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのセーフティ速度を有効にする使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 01 FF 00 DC CA

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0401
変更データ[H]	FF00
エラーチェック[H]	DCCA (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.3 サーボ ON/OFF 《SON》

(1) 機能

サーボON/OFFの制御を行います。

変更データ部をサーボON状態にするとメーカーパラメータの「サーボオン遅延時間」経過後にサーボON状態へと遷移します。ただし以下の条件を満たしている必要があります。

- ・デバイスステータスレジスタ1のEMGステータスビットが0
- ・デバイスステータスレジスタ1の重故障ステータスビットが0
- ・デバイスステータスレジスタ2のイネーブルステータスビットが1
- ・システムステータスレジスタの自動サーボOFF中ステータスビットが0

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,DOへの1点書き込み
開始アドレス[H]	2	0403	サーボON/OFF指令
変更データ[H]	2	任意	サーボ ON: FF00 _H サーボ OFF: 0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※上位との通信前に、ティーチングボックス、パソコンソフトを接続し、サーボ OFF 動作をした後に、非接続にした場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをサーボONにする使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 03 FF 00 7D 0A

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0403
変更データ[H]	FF00
エラーチェック[H]	7D0A (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.4 アラームリセット《ALRS》

(1) 機能

アラームリセットのエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、アラームリセットを行います。

ただしアラーム要因が解消されていないと再びアラームとなります。

また一時停止中にアラームリセットのエッジを立てると残移動量のキャンセルが行われます。

アラームリセットを行ったら 必ず変更データを0000_Hにして書き込みを行い、通常の状態に戻してください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0407	アラームリセット指令
変更データ[H]	2	任意	アラームリセット実行:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのアラームリセットを行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 07 FF 00 3C CB ...アラームリセット実行

2回目 01 05 04 07 00 00 7D 3B ...通常状態に戻す

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0407
変更データ[H]	1回目:FF00 2回目:0000 (アラームリセット終了後に0000 _H を書き込んで通常状態に戻してください)
エラーチェック[H]	1回目:3CCB (CRC計算による) 2回目:7D3B (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 05

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.5 ブレーキ強制解除《BKRL》

(1) 機能

ブレーキの制御はサーボON/OFFと連動して行われますが、ブレーキがONの状態でも強制的に解除することができます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0408	ブレーキ強制解除指令
変更データ[H]	2	任意	ブレーキ強制解除:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※上位との通信前に、パソコン対応ソフトなどのティーチングツールを接続し、サーボOFF動作をした後に、接続を外した場合、上位との通信でサーボON/OFFができなくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのブレーキを強制解除する使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 08 FF 00 0C C8

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0408
変更データ[H]	FF00
エラーチェック[H]	0CC8 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.6 一時停止《STP》

(1) 機能

移動中に一時停止指令を行うと減速停止を行い、再び通常状態にセットされると残移動量の移動を再開します。

一時停止指令の状態では、モータの移動は全て禁止されます。

一時停止指令中にアラームリセット指令ビットが立てられた場合は残移動量がキャンセルされます。

一時停止指令の入力が原点復帰動作中で、押付け反転前ならば移動指令が保留され、押付け反転後では原点復帰を最初からやり直します。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,DOへの1点書き込み
開始アドレス[H]	2	040A	一時停止指令
変更データ[H]	2	任意	一時停止指令:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを一時停止する使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 0A FF 00 AD 08

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	040A
変更データ[H]	FF00
エラーチェック[H]	AD08 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.7 原点復帰《HOME》

(1) 機能

原点復帰指令のエッジを立てる(変更データが0000_Hの状態にFF00_Hを書き込む)と、原点復帰動作を行います。原点復帰が完了するとHENDビットが1になります。

原点復帰指令は、原点復帰が完了していても入力可能です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,DOへの1点書き込み
開始アドレス[H]	2	040B	原点復帰指令
変更データ[H]	2	任意	原点復帰実行:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※原点復帰は、サーボ ON した状態で行ってください。

上位との通信前に、パソコン対応ソフトなどのテスターツールを接続し、サーボ OFF 動作をした後に、接続を外した場合、上位との通信でサーボ ON/OFF ができなくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボ ON の状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラの原点復帰を行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 0B 00 00 BD 38 ...通常状態に設定

2回目 01 05 04 0B FF 00 FC C8 ...原点復帰実行

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	040B
変更データ[H]	1回目:0000 2回目:FF00 (エッジを立てる為に2回データを送信してください。)
エラーチェック[H]	1回目:3CCB (CRC計算による) 2回目:7D3B (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.8 位置決め動作起動指令《CSTR》

(1) 機能

位置決め動作起動指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、ポジション番号指定レジスタ(POSR:0D03_H)内のポジション番号の指定位置に移動します。ポジションスタート指令状態のまま(FF00_Hを書き込んだまま)ですと位置決め幅内に入っても完了ポジションは出力されません。(0000_Hを書き込んで通常状態に戻してください。)

電源投入後、一度も原点復帰動作を行っていない状態(HENDビットが0の状態)では、原点復帰動作を実行した後に目標位置に移動を開始します。

※目標位置 及び速度等の動作パラメータは、全てコントローラ内部のポジションテーブル(不揮発性メモリ)に予め設定しておく必要があります。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	040C	位置決め動作起動指令
変更データ[H]	2	任意	ポジションスタート指令:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのポジション番号指定レジスタ(POSR:0D03_H)内のポジション番号の指定位置に移動を行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 0C FF 00 4D 09 ...指定位置に移動

2回目 01 05 04 0C 00 00 0C F9 ...通常状態に戻す

フィールド ^① 名称	RTUモード ^② 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	040C
変更データ[H]	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック[H]	1回目:4D09 (CRC計算による) 2回目:0CF9 (CRC計算による)
エンド	サイレントインターバル

RTU
ファンクション
コード 05

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.9 ジョグ/インチング切替《JISL》

(1) 機能

ジョグとインチングの切替を行います。

ジョグ動作中に本ビットが切替わると減速停止します。

インチング動作中に本ビットが切替わってもインチング動作は継続されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレープアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0411	ジョグ/インチング切替
変更データ[H]	2	任意	インチング動作状態:FF00 _H ジョグ動作状態:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをインテック動作に切替えます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 11 FF 00 DD 0F

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0411
変更データ[H]	FF00
エラーチェック[H]	DD0F (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。



RTU
ファンクション
コード 05

5.4.10 ティーチモード指令《MOD》

(1) 機能

通常運転モードと教示モードを切替えます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0414	通常モード⇔教示モード切替
変更データ[H]	2	任意	教示モード:FF00 _H 通常運転モード:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを教示モードに切替えます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 14 FF 00 CD 0E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0414
変更データ[H]	FF00
エラーチェック[H]	CD0E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.11 ポジションデータ取込み指令《TEAC》

(1) 機能

5.4.10 テーチモード指令が FF00_H(教示モード)の時に、本指令(FF00_Hを書き込み)で現在位置データの取込みを行います。

取込み場所は、ポジション番号指定レジスタで指定されているポジション番号の中です。取込みポジションが空のポジションの場合、目標位置以外のデータ(位置決め幅 INP,速度 VCMD,加減速度 ACMD,制御フラグ CTLF)はパラメータの初期値と一緒に書込まれます。

本指令(FF00_Hを書き込み)を行って 20ms 以上そのままの状態を保ってください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0415	ポジションデータ取込指令
変更データ[H]	2	任意	ポジションデータ取込み指令:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラが教示モード時に現在位置を取込みます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 15 FF 00 9C CE

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0415
変更データ[H]	FF00
エラーチェック[H]	9CCE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

RTU
ファンクション
コード 05

5.4.12 ジョグ+指令《JOG+》

(1) 機能

ジョグ またはインテグ動作を行います。

・5.4.9 ジョグ/インテグ切替指令が 0000_H(ジョグ設定)の時、ジョグ+指令(変更データ FF00_H)を送信すると、反原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26のPIOジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ+指令(変更データ 0000_H)を送信するか、5.4.13 ジョグ-指令(変更データ FF00_H)を送信すると減速停止します。

・5.4.9 ジョグ/インテグ切替指令が FF00_H(インテグ設定)の時、ジョグ+指令のエッジを立てる(変更データが 0000_Hの状態では FF00_Hを書き込む)と、反原点方向にインテグ移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインテグ距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0416	ジョグ+指令
変更データ[H]	2	任意	ジョグ+指令:FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 16 FF 00 6C CE

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0416
変更データ[H]	FF00
エラーチェック[H]	6CCE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 16 FF 00 6C CE ...インチング移動

2回目 01 05 04 16 00 00 2D 3E ...通常状態に戻す

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0416
変更データ[H]	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック[H]	1回目:6CCE (CRC計算による) 2回目:2D3E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.13 ジョグ指令《JOG-》

(1) 機能

ジョグ またはインチング動作を行います。

・5.4.9 ジョグ／インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ指令(変更データ FF00_H)を送信すると、原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 の PIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ指令(変更データ 0000_H)を送信するか、5.4.12 ジョグ+指令(変更データ FF00_H)を送信すると減速停止します。

・5.4.9 ジョグ／インチング切替指令が FF00_H(インチング設定)の時、ジョグ指令のエッジを立てる(変更データが 0000_Hの状態では FF00_Hを書き込む)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIO ジョグ速度)、ユーザパラメータNo.48(PIO インチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,DOへの1点書き込み
開始アドレス[H]	2	0417	ジョグ指令
変更データ[H]	2	任意	ジョグ指令: FF00 _H 通常:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 17 FF 00 3D 0E

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0417
変更データ[H]	FF00
エラーチェック[H]	3D0E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 17 FF 00 3D 0E ...インチング移動

2回目 01 05 04 17 00 00 7C FE ...通常状態に戻す

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0417
変更データ[H]	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック[H]	1回目:3D0E (CRC計算による) 2回目:7CFE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.14 スタートポジション0～7(ST0～ST7)移動指令 (PIOパターン4, 5 限定)

(1) 機能

指定されたポジションNo.位置に移動します。

スタートポジション0～7 移動指令は、PIOパターン4, 5(電磁弁モード)が選択されている時に使用できます。

移動指令は、5.4.14 (5)開始アドレス内のST0～ST7のどれかを有効にする(0000_Hの状態
でFF00_Hを書き込む)ことで行います。

有効スタートポジション以外を選択するとアラーム「085 移動時ポジションNo.異常」が発生します。

ユーザパラメータNo.27 移動指令種別によりレベル動作とエッジ動作が選択可能です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ～10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	任意	5.4.14(5)開始アドレス参照
変更データ[H]	2	任意	※1 動作指令 ON:FF00 _H 動作指令OFF:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※1 ユーザパラメータNo.27 移動指令種別を『レベル動作』設定した場合

FF00_H→0000_H書き込みで減速停止します。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをスタートポジション2へ移動します。

スタートポジション設定例

No	位置 [mm]	速度 [mm/s]	加速度 [G]	減速度 [G]
0	0.00	533.00	0.30	0.30
1	25.00	533.00	0.30	0.30
2	50.00	533.00	0.30	0.30

図5.2

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 1D 00 00 5C FC ...エッジを立てる為、0000_H書込み

2回目 01 05 04 1D FF 00 1D 0C ...移動指令

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	041D
変更データ[H]	1回目:0000 2回目:FF00
エラーチェック[H]	1回目:5CFC (CRC計算による) 2回目:1D0C (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

(5) 開始アドレス

アドレス	記号	名称	機能
0418	ST7	スタートポジション 7	ポジション 7 へ移動します
0419	ST6	スタートポジション 6	ポジション 6 へ移動します
041A	ST5	スタートポジション 5	ポジション 5 へ移動します
041B	ST4	スタートポジション 4	ポジション 4 へ移動します
041C	ST3	スタートポジション 3	ポジション 3 へ移動します
041D	ST2	スタートポジション 2	ポジション 2 へ移動します
041E	ST1	スタートポジション 1	ポジション 1 へ移動します
041F	ST0	スタートポジション 0	ポジション 0 へ移動します

5.4.15 ロードセルキャリブレーション指令《CLBR》専用ロードセル接続が必要

(1) 機能専用

専用ロードセルのキャリブレーションを行います。

ロードセルは工場出荷時、無負荷の状態を ON とするよう設定していますが、負荷を取付けた状態を基準(ON)としたい場合などには、キャリブレーションを行ってください。

その他にも必要な場合(再調整、点検等)、状況に応じて実施してください。

(2) ケリフォーマット

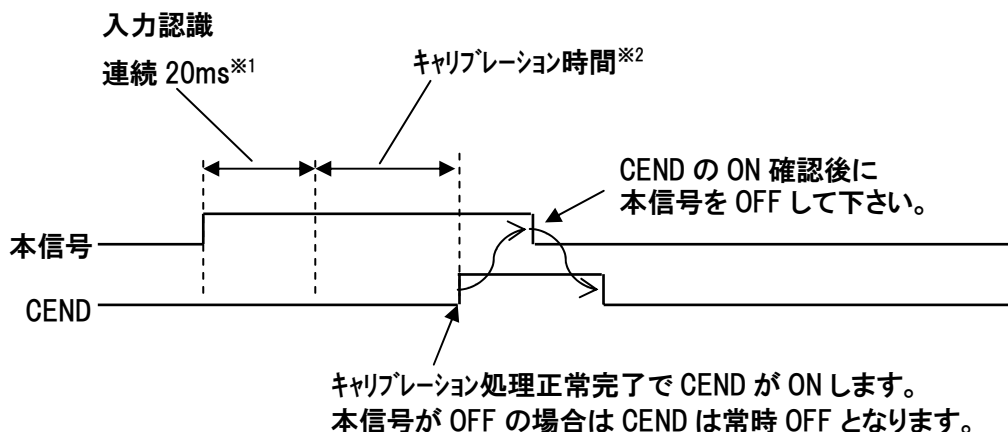
フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	0426	ロードセルキャリブレーション指令
変更データ[H]	2	任意	キャリブレーション実行指令:FF00 _H 通常運転時:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) キャリブレーション手順

- ① 運転を停止してください(軸動作中、押付け中、一時停止中はキャリブレーションできずに0E1:ロードセルキャリブレーション異常アラームとなります)。
- ② 本信号を 20ms 以上連続 ON してください。
- ③ キャリブレーションが完了するとキャリブレーション完了信号(4.3.2(12)デバイスステータスレジスタ1のCEND)が ON しますので、その後本信号を OFF してください。
キャリブレーションが正常に終了しなかった場合、0E1:ロードセルキャリブレーション異常アラームとなります。



注意:本信号が ON 状態では、通常運転指令は受け付けられません。



※1 この間に本信号を OFF した場合は、入力認識前のためキャリブレーション処理を行いません。

※2 この間に本信号を OFF した場合、アラームとなります。

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラに接続された専用ロードセルのキャリブレーションを行います。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 26 FF 00 6C C1

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0426
変更データ[H]	FF00
エラーチェック[H]	6CC1 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5. 4. 16 PIO/Modbus 切替設定 《PMSL》

(1) 機能

PIO 外部指令信号の有効/無効の切替を行うことができます。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレープアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,DOへの1点書き込み
開始アドレス[H]	2	0427	PIO/Modbus切替設定
変更データ[H]	2	任意	※1 Modbus指令有効:FF00 _H Modbus指令無効:0000 _H
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※1 ・Modbus 指令有効(ON)(PIO 指令無効): FF00_H

PIO 信号による運転はできません。

・Modbus 指令無効(OFF)(PIO 指令有効): 0000_H

外部からの PIO 信号による運転が可能です。

補足 Modbus 指令を有効に変更した場合、変更時の PIO 状態が保持されています。

Modbus 指令を無効に切替えた場合、現在の PIO 状態により運転状態が変化します。但し、その時にエッジ検出で動作をする信号の状態が変化していてもエッジを検出したことにしないようにしています。

(3) 注意事項

■運転モードスイッチ搭載機種では AUTO モードに変更されたら『PIO 指令有効』に

MANU モードに変更されたら『PIO 指令無効』になります。

■PIO 未搭載機種ではデフォルト設定が『PIO 指令無効』となります。

■当社ツール接続時(ティーチングペンダント、パソコン対応ソフト)は、ツール内モードには『ティーチモード 1, 2』、『モニターモード 1, 2』が存在します。この場合

『モニターモード 1, 2』 → 『PIO 指令有効』

『ティーチモード 1, 2』 → 『PIO 指令無効』 となります。

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをModbus指令有効にします。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 27 FF 00 3D 01

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	0427
変更データ[H]	FF00
エラーチェック[H]	3D01 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.17 減速停止《STOP》

(1) 機能

減速停止指令のエッジを立てる(FF00_Hを書込む)と、減速停止します。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレープアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	05	コイル,D0への1点書き込み
開始アドレス[H]	2	042C	減速停止設定
変更データ[H]	2	任意	減速停止指令(ON):FF00 _H ※コントローラが自動的に0000 _H に リセットします。
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを減速停止させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 2C FF 00 4C C3

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	05
開始アドレス[H]	042C
変更データ[H]	FF00
エラーチェック[H]	4CC3 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.5 制御情報の直接書き込み(使用ファンクションコード 06)

5.5.1 レジスタへの書き込み

(1) 機能

スレーブのレジスタの内容を変更(書き込み)します。

ブロードキャストの場合には、全スレーブの同じアドレスのレジスタの内容が変更されます。

[4.3.2 (5) デバイス制御レジスタ1 内容参照]

[4.3.2 (6) デバイス制御レジスタ2 内容参照]

[4.3.2 (7) ポジション番号指定レジスタおよびポジション移動指定レジスタ内容参照]

(2) 開始アドレス一覧

アドレス	記号	名称	バイト
0D00	DRG1	デバイス制御レジスタ1	2
0D01	DRG2	デバイス制御レジスタ2	2
0D03	POSR	ポジション番号指定レジスタ	2
9800	POSR	ポジション移動指定レジスタ	2

上記は制御指令のレジスタです。本レジスタのヒットは、「PIO/Modbus切替えステータス(PMSS)[4.3.2(14)参照]」がModbus指令無効(PIO指令有効)の時、PIOパターンにより入力ポートに割り当てられます。本レジスタはModbus指令有効(PIO指令無効)の時、書換えが可能です。

(3) クエリフォーマット

クエリメッセージでは、変更するレジスタのアドレスとデータを指定します。

変更したいデータは、クエリの変更データエリアで16ビットのデータとして指定します。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	06	レジスタへの書込み
開始アドレス[H]	2	任意	5.5.1(2)開始アドレス一覧参照
変更データ[H]	2	任意	4.3.2(5)~4.3.2(7) 変更データ一覧参照
エラーチェック[H]	2	計算結果に基づく	
エンド	なし		サイレントインターバル
合計バイト数	8		

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

以下①～③に、それぞれの運転動作に応じた例を示します。

①軸No.0コントローラをサーボON→原点復帰させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 06 0D 00 10 00 86 A6 ...サーボON

2回目 01 06 0D 00 10 10 87 6A ...原点復帰

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	06
開始アドレス[H]	0D00
変更データ[H]	1回目:1000 2回目:1010 (サーボONのビットはサーボOFF時以外は1のままにする)
エラーチェック[H]	1回目:86A6 (CRC計算による) 2回目:876A (CRC計算による)
エンド	サイレントインターバル

注1 サーボOFFの状態から 変更データを1010_Hとして送信しても原点復帰は行われません。[各RCコントローラ取説記載の起動時のタイミングチャート参照]

注2 前の状態を維持したい場合は、変更がなくても前の状態を送信してください。

使用例のようにサーボONのビットは 原点復帰時も1のままにしておきます。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②ポジション移動指定レジスタ(アドレス9800_H)を使用してポジションNo.1に移動

前例①を行い、原点復帰を完了した状態で行ってください。

クエリ(前後にサイレントインターバルが入ります)

01 06 98 00 00 01 67 6A

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	06
開始アドレス[H]	9800
変更データ[H]	0001
エラーチェック[H]	676A (CRC計算による)
エンド	サイレントインターバル

注 本レジスタにポジションNo.を書き込むと移動を開始します。CSTR(スタート信号)は必要ありません。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

③ポジション番号指定レジスタ(アドレス0D03_H)を使用してポジションNo.1に移動

前例①を行い、原点復帰を完了した状態で行ってください。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 06 0D 03 00 01 BA A6 ...ポジションNo.1を指定

2回目 01 06 0D 00 10 00 86 A6 ...CSTR(スタート信号)OFFを入力

3回目 01 06 0D 00 10 08 87 60 ...CSTR(スタート信号)ONを入力

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス[H]	01
ファンクションコード[H]	06
開始アドレス[H]	1回目:0D03 2回目:0D00 3回目:0D00
変更データ[H]	1回目:0001 2回目:1000 3回目:1008
エラーチェック[H]	1回目:BA A6 (CRC計算による) 2回目:86 A6 (CRC計算による) 3回目:87 60 (CRC計算による)
エンド	サイレントインターバル

注 前の状態を維持したい場合は、変更がなくても前の状態を送信してください。

使用例のようにサーボONのビットはサーボOFF以外では1のままにしてください。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.6 位置決めデータの直接書き込み(使用ファンクションコード 10)

5.6.1 直値移動指令

(1) 機能

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。アドレス 9900_H~9908_H のレジスタ群を書換えるとアクチュエータに対し直値移動指令が行えます。(一伝文で送信が可能です。)

制御フラグ指定レジスタ(アドレス:9908_H)以外のレジスタは電源投入後、1度送信すればその後は有効となりますので、目標位置、位置決め幅、速度、加減速度、押し付け電流制限値、制御指定の変更の必要がない場合、その後は単独変更による実移動指令可能なレジスタ(開始アドレス一覧参照)の書込みだけで直値移動指令が可能となります。

(2) 開始アドレス一覧

目標位置座標、位置決め幅、速度、加減速度、押し付け電流制限値制御指定フラグ等を数値指定して移動を行うためのレジスタ群です。

開始アドレス一覧のデータ(合計 8 レジスタ)は、一度の送信で変更することが可能です。

アドレス [H]	記号	名称	符号	単独変更による 実移動指令 可能	レジスタ サイズ*	バイト サイズ*	単位
9900	PCMD	目標位置指定レジスタ	○	○	2	4	0.01mm
9902	INP	位置決め幅指定レジスタ		×	2	4	0.01mm
9904	VCMD	速度指定レジスタ		○	2	4	0.01mm/s
9906	ACMD	加減速度指定レジスタ		○	1	2	0.01G
9907	PPOW	押し付け時電流制限 指定レジスタ		○	1	2	%
9908	CTLF	制御フラグ指定レジスタ		×	1	2	—

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート		なし	サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	10	直値指令
開始アドレス[H]	2	任意	5.6.1(2)開始アドレス一覧参照
レジスタの数[H]	2	任意	5.6.1(2)開始アドレス一覧参照
バイト数[H]	1	上記レジスタ数による	上記指定レジスタ数の2倍の数値を 入力
変更データ1[H]	2		5.6.1(2)開始アドレス一覧参照
変更データ2[H]	2		5.6.1(2)開始アドレス一覧参照
変更データ3[H]	2		5.6.1(2)開始アドレス一覧参照
:	:		:
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート		なし	サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	10	直値指令
開始アドレス[H]	2	任意	5.6.1(2)開始アドレス一覧参照
レジスタの数[H]	2	任意	5.6.1(2)開始アドレス一覧参照
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(5) レジスタ詳細説明

■ 目標位置指定レジスタ(PCMD)

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。単位は 0.01mm で、設定可能範囲は、-999999~999999(FFF0BDC1^(注 1)~000F423F_H)です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると、ソフトリミット設定値の 0.2mm 手前^(注 2)を目標位置として移動を開始します。目標位置座標指定レジスタ(記号: PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

注 1 負の値は、2の補数で設定してください。

注 2 インデックスモード設定の回転軸の場合はソフトリミット設定値が目標位置となります。

■ 位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって2種類の意味を持ちます。1つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、1~999999(1~000F423F_H)です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

本レジスタを変更しただけでは移動開始は行われません。

■ 速度指定レジスタ(VCMD)

移動速度を指定します。単位は 0.01mm/s で設定範囲は 1~999999(1~000F423F_H)です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタの下位ワードが書換えられると移動開始します。つまり移動中の速度可変を行う場合、本レジスタを書換えることにより実現できます。

■加減速度指定レジスタ(ACMD)

加速度・減速度を指定します。単位は 0.01G で設定範囲は 1~300(1~012C_H) です。ただし、パラメータの最大加速度および最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタが書換えられると移動開始します。つまり移動中の加減速度可変を行う場合、本レジスタを書換えることにより実現できます。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。以下を参照して設定してください。

アクチュエータ型式	押付け可能範囲[%]	設定可能範囲(入力値)[H]
RCS2-RA13R 以外の アクチュエータ	20~70 ^(注)	33~B2
RCS2-RA13R	20~200	33~1FE

注 各アクチュエータによっては設定範囲が異なる場合があります。

[詳しくは当社カタログ、または取扱説明書を参照]

本レジスタが書換えられると移動開始します。つまり押付け動作中の電流制限値可変を行う場合、本レジスタを書換えることにより実現できます。

押付け電流値設定例

●20%設定例

$$255(100\%) \times 0.2(20\%) = 51 \rightarrow 33_{\text{H}}(16 \text{ 進数変換})$$

■制御フラグ指定レジスタ(CTLF)

動作方法を設定します。

押付け動作およびインクリメンタル動作(ピッチ送り)の場合、本レジスタを移動指令のたびに設定してください(1回の移動のたびに初期値で書き換えられます)。

CTLF ビット構成

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
MSB	-	-	NTC1	NTC0	-	-	-	-	MOD1	MOD0	GSL1	GSL0	INC	DIR	PUSH	-	LSB

ビット 1(PUSH) = 0:通常動作(初期値)

1:押付け動作

ビット 2(DIR) = 0:アプローチ動作完了後の押付け動作の方向を正転とします(初期値)。

1:アプローチ動作完了後の押付け動作の方向を逆転とします。

このビットにより、PCMD からの最終停止位置の方向を算出しますので方向を間違えると下図 5.3 のように(2×INP)の幅だけずれた動作になりますので、注意してください。

また、ビット 1 の設定値が 0 の場合はこのビットの設定値は無効です。

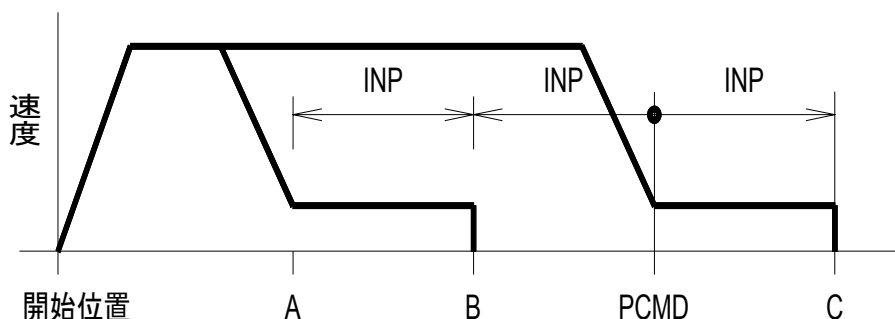


図 5.3 押付け時の動作方向

ビット 3(INC) = 0:通常動作(初期値)

1:インクリメンタル動作(ピッチ送り)

このビットを 1 に設定することにより、相対位置移動を行う事が可能となります。この動作は、通常動作時と押付け動作時(CTLF のビット 1)で挙動が異なります。通常動作時は目標位置(PCMD)に対しての移動量を生成するのに対し、押付け動作時(ビット 1=1 の時)には現在位置に対しての移動量を生成します。

なお、相対座標の計算は mm 単位加算後パルス変換を行うので、パルス変換

後の加算方法の場合に発生する『相対移動を繰り返した場合は、リート設定による割り切れないパルスが累積誤差として位置ずれを起こす現象』は発生しません。

ビット 4(GSL0)、5(GSL1)= 以下の表参照(SCON-CA だけ設定可能)

GSL1	GSL0	機能
0	0	パラメータセット 0 選択(初期値)
0	1	パラメータセット 1 選択
1	0	パラメータセット 2 選択
1	1	パラメータセット 3 選択

あらかじめ 6 種のサーボゲインパラメータを最大 4 セット登録しておくことで、ポジション移動ごとに選択したパラメータセットで運転を行います。[詳細はコントローラ本体の取扱説明書参照]

ビット 6(MOD0)、7(MOD1)= 以下の表参照(PCON-*, ERC2 は設定できません)

MOD1	MOD0	機能
0	0	台形パターン(初期値)
0	1	S 字モーション
1	0	一次遅れフィルタ
1	1	使用できません

加減速パターン特性を選択するための信号です。いずれかをアクチュエータの移動指令前に選択してください。[詳細はコントローラ本体の取扱説明書参照]

ビット 12(NTC0)、13(NTC1)= 以下の表参照(SCON-CA だけ設定可能)

NTC1	NTC0	機能
0	0	制振制御を使用しない(初期値)
0	1	パラメータセット 1 選択
1	0	パラメータセット 2 選択
1	1	パラメータセット 3 選択

制振制御の使用選択 および、あらかじめパラメータを最大 3 セット登録しておくことで、ポジション移動ごとに選択したパラメータセットで運転を行います。[詳細はコントローラ本体の取扱説明書参照]

(6) 使用例

以下①～⑦に、それぞれの運転動作に応じた例を示します。

① 目標位置を変更して移動する(目標位置以外のデータは、パラメータ初期値を使用)

条件:コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値
で動作条件は OK。目標位置だけを変更しアクチュエータを動作させたい。

補足:コントローラユーザパラメータについて

- ・速度初期値(パラメータNo.8)→カタログ内の該当アクチュエータ最高速度
- ・加減速度初期値(パラメータNo.9)→カタログ内の該当アクチュエータ定格加速度
- ・位置決め幅初期値(パラメータNo.10)→デフォルト値 0.1mm

目標位置指定レジスタ(9900_H)の書込み^(例1)



移動開始

(例1) 目標位置を 50mm の位置とする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	設定不要				

■クエリ :01 10 9900 0002 04 0000 1388 38AF

■レスポンス :01 10 9900 0002 6F54

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0002 _H	アドレス9900 _H ～9901 _H まで書込み指定
バイト数	04 _H	2(レジスタ)×2=4(バイト)→4 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1388 _H	50[mm]×100=5000→1388 _H
エラーチェック	38FF _H	CRCチェック計算結果→38FF _H
エンド	なし	サイレントインターバル
合計バイト数	13	

②目標位置を変更して移動する(目標位置以外のデータも変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させたい

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み(例2)

移動開始

(例2) 目標位置を 50mm の位置とする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	0.1	100	0.3		設定不要

■クエリ :01 10 9900 0007 0E 0000 1388 0000 000A 0000 2710 001E 50CF

■レスポンス :01 10 9900 0007 AF57

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0007 _H	アドレス9900 _H ～9906 _H まで書込み指定
バイト数	0E _H	7(レジスタ)×2=14(バイト)→E _H
変更データ1、2(目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1388 _H	50[mm]×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1[mm]×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100[mm/s]×100=10000→2710 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3[G]×100=30→001E _H
エラーチェック	50CF _H	CRCチェック計算結果→50CF _H
エンド	なし	サイレントインターバル
合計バイト数	23	

③移動中の速度変更を行う

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み^(例 2)



移動開始



速度指定レジスタ(9904_H～9905_H)の書込み^(例 3)



変更された速度で引き続き移動



RTU
ファンクション
コード 10

(例3) 移動中に速度 100mm/s→50mm/s に変更する場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	0.1	100→50	0.3		設定不要

(1)速度 100mm/s で移動開始[前例の②目標位置を変更して移動するを参照]

■クエリ :01 10 9900 0007 0E 0000 1388 0000 000A 0000 2710 001E 50CF

■レスポンス :01 10 9900 0007 AF57

(2)速度 50mm/s に速度変更

■クエリ :01 10 9904 0002 04 0000 1388 395C

■レスポンス :01 10 9904 0002 2E95

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳(50mm/s に速度変更 [100mm/s で移動開始は前例参照])

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9904 _H	開始アドレスは目標位置指定レジスタ9904 _H から
レジスタの数	0002 _H	アドレス9904 _H ～9905 _H まで書込み指定
バイト数	04 _H	2(レジスタ)×2=4(バイト)→4 _H
変更データ5、6(速度) 入力単位(0.01mm/s)	0000 _H	32ビットデータ上位ビットは全て0
	1388 _H	50[mm/s]×100=5000→1388 _H
エラーチェック	395C _H	CRCチェック計算結果→395C _H
エンド	なし	サイレントインターバル
合計バイト数	13	

④インクリメンタル移動(ピッチ送り)の場合

条件:コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値
/で動作条件は OK。ピッチ幅だけを変更しアクチュエータを動作させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例 4)



移動開始

補足:アドレス 9900_Hと 9908_Hだけを一度のデータ送信で変更することはできません。アドレスは連番となっているため、9900_Hと 9908_Hだけを変更したい場合は、2 度の伝文送信で実施してください。

1 度の伝文送信で済ませたい場合は、9900_H～9908_H までを全て書込み実行を行ってください。

(例4) ピッチ幅を 10mm に設定してインクリメンタル移動する場合

ピッチ幅 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
10	0.1	100	0.3	0	インクリメンタル (bit3=1)

■クエリ :01 10 9900 0009 12 0000 03E8 0000 000A 0000 2710 001E 0000 0008 F3A0

■レスポンス :01 10 9900 0009 2E93

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0009 _H	アドレス9900 _H ~9908 _H まで書込み指定
バイト数	12 _H	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	03E8 _H	10[mm]×100=1000→03E8 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1[mm]×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/s)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100[mm/s]×100=10000→2710 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	0000 _H	0[%]→0 _H
変更データ9(制御フラグ)	0008 _H	(インクリメンタル設定) 1000b→0008 _H
エラーチェック	F3A0 _H	CRCチェック計算結果→F3A0 _H
エンド	なし	サイレントインターバル
合計バイト数	27	

⑤インクリメンタル移動(ピッチ送り)中に速度変更を行う場合

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを相対位置決め動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例4)



インクリメンタル移動開始



速度指定レジスタ(9904_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例5)



変更された速度で引き続きインクリメンタル移動

補足: 制御フラグ指定レジスタ(9908_H)は設定後、一度アクチュエータが動作開始すると初期値(0_H:通常移動)に戻ります。このため、インクリメンタルおよび押付け動作を再度行う場合は、制御フラグ指定レジスタ(9908_H)を再設定し、送信する必要があります。

(例5) 移動中に速度 100mm/s→50mm/s に変更する場合

ピッチ幅 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
10	0.1	100→50	0.3	0	インクリメンタル (bit3=1)

(1)速度 100mm/s で移動開始[前例の④インクリメンタル移動(ピッチ送り)の場合を参照]

■クエリ :01 10 9900 0009 12 0000 03E8 0000 000A 0000 2710 001E 0000 0008 F3A0

■レスポンス :01 10 9900 0009 2E93

(2)速度 50mm/s に速度変更

■クエリ :01 10 9904 0005 0A 0000 1388 001E 0000 0008 BD83

■レスポンス :01 10 9904 0005 6F57

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳(50mm/s に速度変更 [100mm/s で移動開始は前例参照])

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9904 _H	開始アドレスは目標位置指定レジスタ9904 _H から
レジスタの数	0005 _H	アドレス9904 _H ～9908 _H まで書込み指定
バイト数	0A _H	5(レジスタ)×2=10(バイト)→A _H
変更データ5、6(速度) 入力単位(0.01mm/s)	0000 _H 1388 _H	32ビットデータ上位ビットは全て0 50[mm/s]×100=5000→1388 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	0000 _H	0[%]→0 _H
変更データ9(制御フラグ)	0008 _H	(インクリメンタル設定) 1000b→0008 _H
エラーチェック	BD83 _H	CRCチェック計算結果→BD83 _H
エンド	なし	サイレントインターバル
合計バイト数	19	

⑥押付け動作の場合(押付け動作中の押付け力の変更)

条件: 押付け動作をさせたい。但し、押し付け中には任意のタイミングで押付け力を変化させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:
押付け設定)の書込み^(例6)



押付け動作開始



押付け動作中に押付け電流制限指定レジスタ(9907_H)～制御
フラグ指定レジスタ(9908_H:押付け設定)の書込み^(例7)



変更された押付け力で引き続き押付け動作

(例6) 50mmの位置から20mmの幅で押付け電流制限70%の押付け動作をする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	20	100	0.3	70	押付け動作 (bit1=1、 Bit2=0、1)

■クエリ :01 10 9900 0009 12 0000 1388 0000 07D0 0000 2710 001E 00B2 0006 C377

■レスポンス :01 10 9900 0009 2E93

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0009 _H	アドレス9900 _H ~9908 _H まで書込み指定
バイト数	12 _H	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1388 _H	50[mm]×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	07D0 _H	20[mm]×100=2000→07D0 _H
変更データ5、6(速度) 入力単位[0.01mm/s]	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100[mm/s]×100=10000→2710 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	00B2 _H	70[%]→B2 _H
変更データ9(制御フラグ)	0006 _H	(押付け設定) 0110b→0006 _H
エラーチェック	C377 _H	CRCチェック計算結果→C377 _H
エンド	なし	サイレントインターバル
合計バイト数	27	

(例7) 押付け動作中に押付け電流制限を70%→50%に変更する場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	20	100	0.3	70→50	押付け動作 (bit1=1、 bit2=1)

■クエリ :01 10 9907 0002 04 007F 0006 C5C5

■レスポンス :01 10 9907 0002 DE95

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9907 _H	開始アドレスは目標位置指定レジスタ9907 _H から
レジスタの数	0002 _H	アドレス9907 _H ～9908 _H まで書込み指定
バイト数	04 _H	2(レジスタ)×2=4(バイト)→4 _H
変更データ8(押付け) 入力単位[%]	007F _H	50[%]→7F _H
変更データ9(制御フラグ)	0006 _H	(押付け設定) 0110b→0006 _H
エラーチェック	C5C5 _H	CRCチェック計算結果→C5C5 _H
エンド	なし	サイレントインターバル
合計バイト数	13	

RTU
ファンクション
コード 10

⑦注意事項(移動中の位置決め幅の変更)

移動中の位置決め幅の変更はできません。

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングで位置決め幅を変更をしたい
(変更できません。書込みを行った場合、このデータは次の位置決めに反映されることとなります。)

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み



通常動作開始



位置決め幅指定レジスタ(9902_H～9903_H)の書込み



変更前の位置決め幅設定で引き続き通常動作

補足: 位置決め幅指定レジスタは単独書込みによる実移動指令は無効です。
このため位置決め幅指定レジスタ(9902_H～9903_H)の書込みによるデータは次回移動指令を実施した際に有効となります。

5.6.2 ポジションテーブルデータ書込み

(1) 機能

このクエリを使用することによってポジションテーブル上のデータ変更が可能です。

開始アドレス一覧(アドレス+0000_H～+000E_H)にアクセスがある毎に 1 ポジションデータ単位で不揮発性メモリ(EEPROM、FeRAM)から読出され、書込み実施後、不揮発性メモリに再び格納されます。各コントローラ取扱説明書の基本仕様から書込み回数制限を確認してください。

※EEPROMはデバイスの制約上、書込み回数が約10万回と制限されています。ポジションテーブルデータの書換えを頻繁に行うと短時間でEEPROMの書換え回数をオーバーし故障の原因となりますので、上位側のロジックは想定外のループ等が発生しないようご注意ください。

FeRAMは書込み回数に制限がありません。

(2) 開始アドレス一覧

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_{\text{H}} + (16 \times \text{ポジション No.})_{\text{H}} + \text{アドレス(オフセット値)}_{\text{H}}$$

例 ポジションNo.200 の速度指令レジスタを変更したい場合

$$\begin{aligned} & 1000_{\text{H}} + (16 \times 200 = 3200)_{\text{H}} + 4_{\text{H}} \\ & = 1000_{\text{H}} + \text{C80}_{\text{H}} + 4_{\text{H}} \\ & = 1\text{C84}_{\text{H}} \end{aligned}$$

『1C84』がクエリ開始アドレス部入力値になります。

注 最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

■ポジションデータ変更レジスタ群

アドレス	記号	名称	符号	レジスタサイズ*	バイトサイズ*	入力単位
+0000	PCMD	目標位置	○	2	4	0.01mm
+0002	INP	位置決め幅		2	4	0.01mm
+0004	VCMD	速度指令		2	4	0.01mm/s
+0006	ZNMP	個別ゾーン境界+側	○	2	4	0.01mm
+0008	ZNLP	個別ゾーン境界-側	○	2	4	0.01mm
+000A	ACMD	加速度指令		1	2	0.01G
+000B	DCMD	減速度指令		1	2	0.01G
+000C	PPOW	押付け時電流制限値		1	2	%
+000D	LPOW	負荷電流閾値		1	2	%
+000E	CTLF	制御フラグ指定		1	2	

※ '+' が付いているアドレスはオフセット値です。

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	10	直値指令
開始アドレス[H]	2	任意	5.6.2(2)開始アドレス一覧参照
レジスタの数[H]	2	任意	5.6.2(2)開始アドレス一覧参照
バイト数[H]	1	上記レジスタ数による	上記指定レジスタ数の2倍の数値 を入力
変更データ1[H]	2		5.6.2(2)開始アドレス一覧参照
変更データ2[H]	2		5.6.2(2)開始アドレス一覧参照
変更データ3[H]	2		5.6.2(2)開始アドレス一覧参照
:	:		:
エラーチェック[H]		CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス[H]	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	1	10	直値指令
開始アドレス[H]	2	任意	5.6.2(2)開始アドレス参照一覧
レジスタの数[H]	2	任意	5.6.2(2)開始アドレス参照一覧
エラーチェック[H]	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(5) レジスタ詳細説明

■目標位置(PCMD)

ポジション移動時の位置決め目標位置を絶対座標上の位置、または相対距離で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999(FFF0BDC1^(注)～000F423F_H)です。絶対座標指定時はパラメータのソフトリットを超えた値が設定されていると、ソフトリット設定値の 0.2mm 手前^(注2)を目標位置として移動を開始します。目標位置座標指定レジスタ(記号:PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

注 1 負の値は、2の補数で設定してください。

注 2 インデックスモード設定の回転軸の場合はソフトリット設定値が目標位置となります。

■位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって2種類の意味を持ちます。1つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、1～999999 (1～000F423F_H)です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■速度指定レジスタ(VCMD)

ポジション移動時の、移動速度を指定します。単位は 0.01mm/sec で設定範囲は 1～999999(1～000F423F_H) です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■個別ゾーン境界±(ZNMP,ZNLP)

パラメータで設定されるゾーン境界とは別に、ポジション移動時だけ有効なゾーン信号を出力します。

絶対位置座標で表現した+側のゾーン信号出力境界値を ZNMP に、-側のゾーン信号出力境界値を ZNLP に設定します。現在位置がこの±境界値の内側にあるときは、ゾーンステータスレジスタの対応するビットが ON となります。設定単位は 0.01mm となります。設定可能範囲はともに -999999～999999(FFF0BDC1^(注)～000F423F_H) となります。個別ゾーン出力を無効にする場合は、ZNMP と ZNLP を同じ値としてください。

注 負の値は、2の補数で設定してください。

■加速度指定レジスタ(ACMD)

ポジション移動時の、加速度を指定します。単位は 0.01G で設定範囲は 1～300 (1～

012C_H) です。ただし、パラメータの最大加速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■減速度指定レジスタ(DCMD)

ポジション移動時の、減速度を指定します。単位は 0.01G で設定範囲は 1~300 (1~012C_H) です。ただし、パラメータの最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。以下を参照して設定してください。

アクチュエータ型式	押付け可能範囲[%]	設定可能範囲(入力値)[H]
RCS2-RA13R 以外の アクチュエータ	20~70 ^(注)	33~B2
RCS2-RA13R	20~200	33~1FE

注 各アクチュエータによっては設定範囲が異なる場合があります。

[詳しくは当社カタログ、または取扱説明書を参照]

押付け電流値設定例

●20%設定例

$255(100\%) \times 0.2(20\%) = 51 \rightarrow 33_{\text{H}}$ (16 進数変換)

■負荷出力電流閾値(LPOW)

負荷出力判定を行う場合、電流閾値を LPOW に設定します。設定は、押付け時電流制限値(PPOW)同様に使用するアクチュエータに合わせて行います。判定を行わない場合は 0 を設定してください。

■制御フラグ指定レジスタ(CTLF)

[5.6.1(5) 制御フラグ指定レジスタ参照]

(6) 使用例

軸No.0のポジションNo.12全データを以下の様書き換えます。

目標位置 [mm]	位置 決め幅 [mm]	速度 [mm/s]	個別ゾーン 境界+側 [mm]	個別ゾーン 境界-側 [mm]	加速度 [G]	減速度 [G]	押付け [%]	閾値	移動制御
100	0.1	200	60	40	0.01	0.3	0	0	通常移動

■クエリ（前後にサイレントインターバルが入ります）

```
01 10 10 C0 00 0F 1E 00 00 27 10 00 00 00 0A 00 00 4E 20 00 00 17 70 00 00 0F A0 00 01
00 1E 00 00 00 00 00 00 70 1E
```

■受信レスポンス 01 10 10 C0 00 0F 84 F1

・・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	10C0 _H	開始アドレスはポジションNo.12における目標位置指定レジスタ10C0 _H から※1
レジスタの数	000F _H	レジスタ記号PCMD~CTLFまで 合計15レジスタ書込み指定
バイト数	1E _H	15(レジスタ)×2=30(バイト)→1E _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100[mm]×100=10000→2710 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1[mm]×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	0000 _H	32ビットデータ上位ビットは全て0
	4E20 _H	200[mm/s]×100=20000→4E20 _H
	0FA0 _H	40[mm]×100=4000→0FA0 _H

次ページに続く

前ページからの続き

フィールド名称	RTUモード 8ビットデータ	備考
変更データ7、8 (個別ゾーン境界+) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1770 _H	60[mm]×100=6000→1770 _H
変更データ9、10 (個別ゾーン境界-) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	0FA0 _H	40[mm]×100=4000→0FA0 _H
変更データ11(加速度) 入力単位(0.01G)	0001 _H	0.01[G]×100=1→0001 _H
変更データ12(減速度) 入力単位(0.01G)	001E _H	0.3[G]×100=30→001E _H
変更データ13(押付け) 入力単位[%]	0000 _H	0[%]→0 _H
変更データ14(閾値) 入力単位[%]	0000 _H	0[%]→0 _H
変更データ15(制御フラグ)	0000 _H	通常動作のため何れのビットも0 0000 _b →0000 _H
エラーチェック	701E _H	CRCチェック計算結果→701E _H
エンド		サイレントインターバル
合計バイト数	39	

※1) 開始アドレス計算

例 ポジションNo.12 の全データを変更をかける為、本クエリの開始アドレス部には
ポジションNo.12 の目標位置アドレスとなる。

$$1000_{\text{H}} + (16 \times 12 = 192)_{\text{H}} + 0_{\text{H}}$$

$$= 1000_{\text{H}} + \text{C0}_{\text{H}} + 0_{\text{H}}$$

$$= 10\text{C0}_{\text{H}}$$

『10C0』が本クエリ開始アドレス部入力値になります。

以下は当社 RC 用パソコンソフト上ポジションデータ内で、クエリメッセージ送信前と後の違いを表示したものです。

(注)パソコンソフトと Modbus の両方共を同時に接続することはできません。次の例は、パソコンソフトと Modbus の接続を切り替えて行った例です。

■クエリ送信前

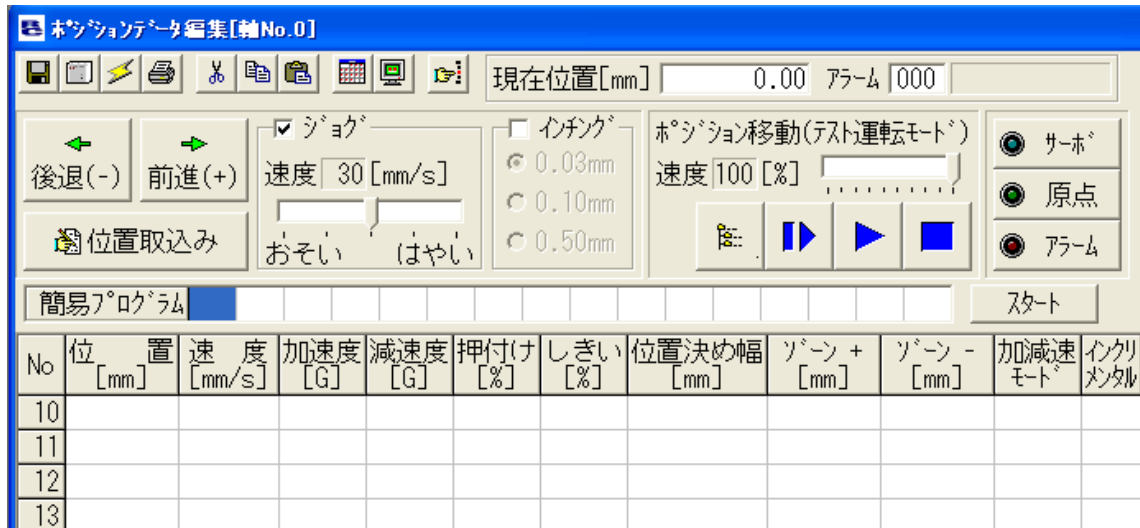


図 5.4

■送信後

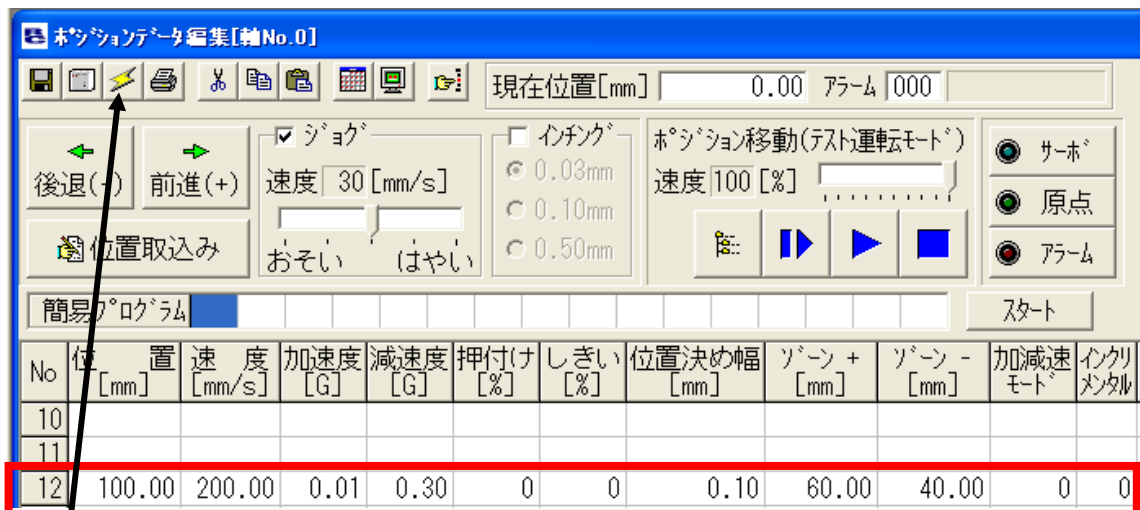



図 5.5

※  ボタンを押すか、ポジションデータ編集画面を開き直さないと書き換えた内容は表示されません。



6 Modbus ASCII



6.1 メッセージフレーム(クエリ、レスポンス)

スタート	アドレス	ファンクションコード	データ	LRC チェック	エンド
1 文字	2 文字	2 文字	n 文字	2 文字	2 文字
1 byte	2 byte	2 byte	n×2 byte	2 byte	2 byte

※ASCIIコードでは、1文字は1バイト(2文字)で表されます。[6.2 ASCIIコード表を参照]

(1) スタート

ヘッダに相当し、ASCIIモードでは‘:’ (コロン)を使用します。

ASCIIコードでは、3A_Hと表されます。

(2) アドレス

接続されているRCコントローラのアドレス(01_H~10_H)を指定します。

アドレス = 軸番号 + 1

をASCIIコードで設定します。 例)軸番号1は、30_H32_Hとなります。

⚠ 注意 アドレスは軸番号と同じでないため 設定にご注意ください。

(3) ファンクション

RCコントローラで使用可能なファンクションコード及び機能を示します。

コード		名称	機能
(Hex)	(ASCII)		
01 _H	30 _H 31 _H	Read Coil Status	コイル、DOの読出し
02 _H	30 _H 32 _H	Read Input Status	入力ステータス、DIの読出し
03 _H	30 _H 33 _H	Read Holding Registers	保持レジスタの読出し
04 _H	30 _H 34 _H	Read Input Registers	入力レジスタの読出し
05 _H	30 _H 35 _H	Force Single Coil	コイル、DOへの1点書込み
06 _H	30 _H 36 _H	Preset Single Register	保持レジスタへの書込み
07 _H	30 _H 37 _H	Read Exception Status	例外ステータス読出し
0F _H	30 _H 46 _H	Force Multiple Coils	複数コイル、DOへの一括書込み
10 _H	31 _H 30 _H	Preset Multiple Registers	複数保持レジスタへの一括書込み
11 _H	31 _H 31 _H	Report Slave ID	スレーブID問い合わせ
17 _H	31 _H 37 _H	Read / Write Registers	レジスタへの読出し、書込み

注 本書では□マークのファンクションコードを使用しています。

注 ROBONETゲートウェイは、ASCIIモードをサポートしていません。

(4) データ

ファンクションコードで指示されたデータを付加する場合に用います。ファンクションコードでデータ付加の指示が無い場合は、データ無しも許されます。

(5) LRC チェック

ASCII モードでは、メッセージに LRC 方式に基づいて最初の colon と CR/LF を除いたメッセージの中身をチェックするエラーチェックが自動的(*)に付加されます。また、チェックはメッセージ中の個別の文字(キャラクタ)のパリティチェック方式と関連せず行われます。

LRC チェックは 2 文字の ASCII コードで構成されています。LRC 値は、LRC をメッセージに付加する送信側が計算します。受信側は、メッセージ受信中に LRC を再計算して、その計算結果と送られてきた値とを比較します。もし、この 2 つの値が一致しなければ、結果はエラーとなります。

* ホスト側は、LRC の計算を行う関数を作成する必要があります。

●〈LRC チェックの計算例〉領域がエラーチェック対象範囲

メッセージクエリが [":"] ["01"] ["05"] ["040B"] ["0000"] [LRC] [CR] [LF] の場合、

①最初に 1 バイト単位で数値を全て加算します。

$$\text{全加算値} = 01_{\text{H}} + 05_{\text{H}} + 04_{\text{H}} + 0B_{\text{H}} + 00_{\text{H}} + 00_{\text{H}} = 15_{\text{H}}$$

②次に 8 ビット長で この値に 2 の補数をとると FFFFFFFE_H となり

LRC は最下位の 1 バイトを除きます。従って LRC = "EB" となります。

(6) エンド

トレー(デリミタ)に相当し、ASCII モードでは "CR/LF" を使用します。

ASCII コードでは、0D_H、0A_H と表されます。

(7) ブロードキャスト

アドレスを 00_H で指定すると接続されている全ての軸に 同一内容のクエリを送信することができます。この場合 RC コントローラからレスポンスは返信されません。

また、本機能は使用できるファンクションコード等に制限がありますので、十分注意してご利用ください。

使用できるファンクションコードは、「6.3 ASCII モード クエリー一覧」でご確認ください。

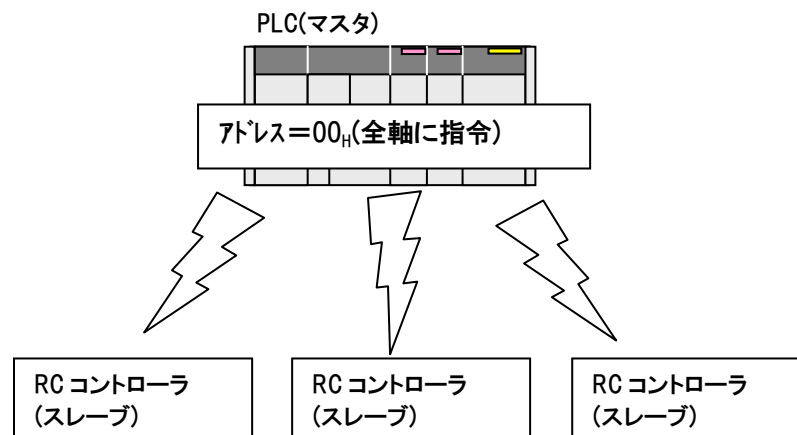


図 6.1

⚠ 注意

・RCコントローラの送受信バッファサイズは、それぞれ 256 バイトとなっています。ホスト側から送信する伝文は受信バッファを、データをリクエストする場合は送信バッファを、それぞれオーバーしないように計算してください。

・データ数が奇数バイトになる場合には以下の理由から注意が必要です。

Modbus 通信の通信データはバイト単位です。

マスタ側では、データを 2 バイト単位で扱っている場合が多く、データ数が奇数になった場合には、自動的に 00_H<NULL>がパケットの最後に付加されてしまう場合があります。

RCコントローラは、マスタ側のインタフェースに Modbus RTU を使用して頂くことを基本にしています。通常 RTU モードで受信待機し、受信後 ASCII かどうかの判別を行っていますので、ヘッダ/データリミットの管理が行えません。

したがって、このような場合には、ASCII モードでの通信ができなくなります。

例)軸No.0 の出力ポート照会の場合

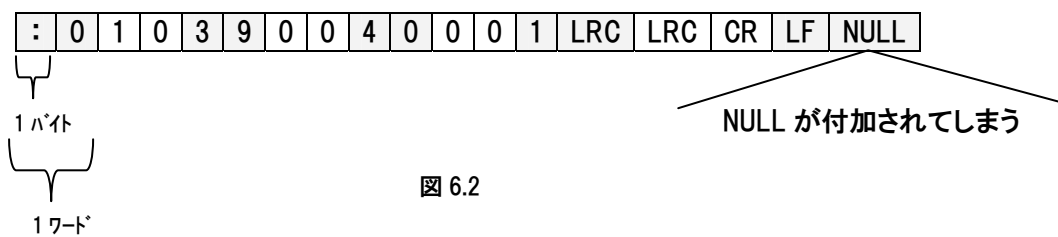


図 6.2

6.2 ASCIIコード表

ASCIIコード（**□**で囲まれた数字や文字(キャラクタ)を変換して伝送します）

上位 3bit 下位 4bit	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P		p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	IS4	,	<	L	¥	l	
D	CR	IS4	-	=	M]	m	}
E	SO	IS4	.	>	N	^	n	
F	SI	IS4	/	?	O	_	o	DEL

-NUL:空文字

-ETX:テキスト終了

-ACK:肯定応答

-HT:水平タブ

-FF:改ページ

-SI:シフトイン

-NAK:否定応答

-CAN:取り消し

-ESC:拡張

-SOH:ヘディング開始

-EOT:伝送終了

-BEL:ベル

-LF:改行

-CR:復帰

-DLE:データリンクでの拡張

-SYN:同期文字

-EM:媒体終端

-SP:スペース

-STX:テキスト開始

-ENQ:問い合わせ

-BS:バックスペース

-VT:垂直タブ

-SO:シフトアウト

-DC*:制御装置*

-ETB:伝送ブロック終了

-DEL:削除

例) 「1」は、ASCIIコードで 31_H → 2進数表記では「00110001」

6.3 ASCIIモード クエリ一覧

FC:ファンクションコード

PIO:パラレル I/O(I/O コネクタの入出力)

PIOとの併用欄、ブロードキャスト欄の○印は、PIOとの併用、ブロードキャストが有効なクエリを表しています

FC	機能	記号	機能概要	PIOとの併用	ブロードキャスト	頁
03	ファンクションコード 03 複数レジスタ読み込み	無し	ファンクション 03 を使用するレジスタを連続的に複数読出すことが可能です。	○		193
03	アラーム詳細内容 の読み取り	ALA0 ALCO ALTO	最後に発生したアラームコード、アラームアドレス、詳細コード、アラーム発生時刻(経過時間)を読取ります。	○		197
03	ポジションデータ の読み取り	右記 参照	指定したNo.のポジションデータを読取ります。 (PCMD,INP,VCMD,ZNMP,ZNLP,ACMD,DCMD,PPOW, LPOW,CTLF)	○		199
03	通算移動回数 の読み取り	TLMC	通算移動回数を読取ります。	○		202
03	通算走行距離 の読み取り	ODOM	通算走行距離を 1m 単位で読取ります。	○		204
03	現在時刻の読み取り	TIMN	現在時刻を読取ります。(PCON-CA/CFA、SCON-CA 専用)	○		206
03	ファン通算駆動時間 の読み取り	TFAN	ファンの通算駆動時間を読取ります。(PCON-CFA 専用)	○		210
03	現在位置の読み取り	PNOW	アクチュエータの現在位置を 0.01mm 単位で読出します	○		212
03	現在発生アラームコード の読み取り	ALMC	現在発生中のアラームコードを読出します。	○		214
03	I/O ポート入力状態 の読み取り	DIPM	PIO 入力ポート ON/OFF 状態を読出します。	○		216
03	I/O ポート出力状態 の読み取り	DOPM	PIO 出力ポート ON/OFF 状態を読出します。	○		220
03	コントローラ状態信号 の読み取り 1 (デバイスステータス 1) (運転準備ステータス)	DSS1	次の 14 項目の状態(ステータス)を読出します。 ① 非常停止 ② セーフティー速度有効/無効 ③ コントローラレディ ④ サーボ ON/OFF ⑤ 押付け空振り ⑥ 重故障 ⑦ 軽故障 ⑧ アブソエラー ⑨ ブレーキ ⑩ 一時停止 ⑪ 原点復帰完了 ⑫ 位置決め完了 ⑬ ロードセルキャリブレーション完了 ⑭ ロードセルキャリブレーションステータス	○		224

03	コントローラ状態信号 の読み取り 2 (デバイスステータス 2) (運転情報 1 ステータス)	DSS2	次の 15 項目の状態(ステータス)を読出します。 ① イネーブル ② 負荷出力判定(検定範囲負荷電流閾値) ③ トルクレベル(負荷電流閾値) ④ ティーチモード(通常/ティーチ) ⑤ ポジションデータ取込(通常/完了) ⑥ ジョグ+(通常/指令中) ⑦ ジョグ-(通常/指令中) ⑧ 完了ポジション 7 ⑨ 完了ポジション 6 ⑩ 完了ポジション 5 ⑪ 完了ポジション 4 ⑫ 完了ポジション 3 ⑬ 完了ポジション 2 ⑭ 完了ポジション 1 ⑮ 完了ポジション 0	○		226
03	コントローラ状態信号 の読み取り 3 (拡張デバイス ステータス) (運転情報 2 ステータス)	DSSE	次の 9 項目の状態(ステータス)を読出します。 ① 非常停止(非常停止入力ポート) ② モータ電圧低下 ③ 運転モード(AUTO/MANU) ④ 原点復帰中 ⑤ 押付け動作中 ⑥ 励磁検出 ⑦ PIO/Modbus 切替 ⑧ ポジションデータ書込み完了ステータス ⑨ 移動中	○		228
03	コントローラ状態信号 の読み取り 4 (システムステータス) (コントローラステータス)	STAT	次の 7 項目の状態(ステータス)を読出します。 ① 自動サーボ OFF 中 ② 不揮発メモリアクセス中 ③ 運転モード(AUTO/MANU) ④ 原点復帰完了 ⑤ サーボ ON/OFF ⑥ サーボ指令 ⑦ 駆動源 ON(通常/しゃ断中)	○		230
03	現在速度 の読み取り	VNOW	アクチュエータの現在速度を 0.01mm/sec 単位で読出します	○		232
03	電流値の読み取り	CNOW	アクチュエータモータの電流指令値を 1mA 単位で読出します	○		234
03	偏差の読み取り	DEVI	1ms 周期毎の偏差量を 1pulse 単位で読出します	○		236
03	電源投入後の積算 時間の読み取り	STIM	コントローラ電源投入時からの積算時間を 1msec 単位で読出します。	○		238

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
03	特殊入力ポートの入力信号状態の読み取り (センサ入力ステータス)	SIPM	次の 8 項目の状態(ステータス)を読み出します。 ① 指令パルス NP ② 指令パルス PP ③ モードスイッチ ④ ベルト切断センサ ⑤ 原点確認センサ ⑥ オーパトラベルセンサ ⑦ クリープセンサ ⑧ リミットセンサ	○		240
03	ゾーン出力信号の読み取り	ZONS	次の 6 項目の状態(ステータス)を読み出します。 ① LS2(PIO パターン電磁弁モード[3 点タイプ]) ② LS1(PIO パターン電磁弁モード[3 点タイプ]) ③ LS0(PIO パターン電磁弁モード[3 点タイプ]) ④ ポジションゾーン ⑤ ゾーン 2 ⑥ ゾーン 1	○		242
03	位置決め完了ポジションNo.の読み取り	POSS	次の 9 項目の状態(ステータス)を読み出します。 ① 完了ポジション番号ビット 256 ② 完了ポジション番号ビット 128 ③ 完了ポジション番号ビット 64 ④ 完了ポジション番号ビット 32 ⑤ 完了ポジション番号ビット 16 ⑥ 完了ポジション番号ビット 8 ⑦ 完了ポジション番号ビット 4 ⑧ 完了ポジション番号ビット 2 ⑨ 完了ポジション番号ビット 1	○		244
03	コントローラ状態信号の読み取り 5	SSSE	次の 2 項目の状態(ステータス)を読み出します。 ① コールドスタートレベルアラームの発生/未発生 ② RTC(カレンダー)機能の使用/未使用 (ERC3、PCON-CA/CFA 限定)	○		246
03	カフィードバックデータの読み取り	FBFC	現在のロードセルの測定値を 0.01N 単位で読み出します。	○		248
05	セーフティ速度有効/無効切替	SFTY	有効/無効モード切替を指令します。		○	252
05	サーボ ON/OFF	SON	サーボ ON/OFF を指令します。		○	254
05	アラームリセット	ALRS	アラームリセット/残移動量キャンセルを指令します。		○	256
05	ブレーキ強制解除	BKRL	ブレーキ強制解除を指令します。		○	258
05	一時停止	STP	一時停止を指令します。		○	260
05	原点復帰	HOME	原点復帰動作を指令します。		○	262

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
05	位置決め動作起動指令	CSTR	ポジションNo.指定移動時のスタート信号です。		○	264
05	ジヨグ/インチング切替	JISL	ジヨグ/インチングモード切替を行います。		○	266
05	ティーチモード指令	MOD	通常/教示モード切替を行います。		○	268
05	ポジションデータ取込み指令	TEAC	教示モード時、現在位置取込指令を行います		○	270
05	ジヨグ+指令	JOG+	反原点方向にジヨグ/インチング動作を指令します。		○	272
05	ジヨグ-指令	JOG-	原点方向にジヨグ/インチング動作を指令します。		○	274
05	スタートポジション 0~7 《ST0~ST7》 移動指令	ST0~ ST7	電磁弁モード時に有効なポジションNo.指定 この指令だけ ^(注) でアクチュエータが動作可能です。		○	276
05	ロードセルキャリブレーション 指令	CLBR	ロードセルのキャリブレーションを行います。		○	278
05	PIO/Modbus 切替設定	PMSL	PIO 外部指令信号の有効/無効切替指令		○	280
05	減速停止	STOP	移動中のアクチュエータを減速停止させる事が出来ます。		○	282
06	制御情報の直接書き込み		コントローラのレジスタの内容を変更(書き込み)します。		○	285
10	直値移動指令	無し	目標位置、位置決め幅、速度、加減速度、押付け、制御設定を1つの伝文で送信し、動作させる事が可能です。通常移動、相対移動、押付け動作が可能です。		○	289
10	ポジションテーブルデータ 書き込み	無し	指定された軸、ポジションNo.のデータを全て変更することが可能です。		○	307
不定	例外レスポンス	無し	伝文内容が不正データであった場合のレスポンスです。			316

6.4 データ、ステータス読み取り(使用ファンクションコード 03)

6.4.1 レジスタの連続複数読み取り

※)6.2 ASCIIコード表
を参照ください。

(1) 機能

スレーブの保持レジスタの内容を読み取ります。

ブロードキャストではサポートされていません。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、ASCIIモード使用時256バイト中9バイト(ヘッダ+スレーブアドレス+ファンクションコード+エラーチェック+トレーラ)を除く247バイト分最大123レジスタ分(1レジスタ2バイト使用)のデータ照会が可能ですので、連続したアドレスの複数レジスタを一度の送受信で照会することも可能です。

アドレス[H]	記号	名称	符号	レジスタサイズ	バイト
0500	ALAO	アラーム詳細コード		1	2
0501	ALAO	アラームアドレス		1	2
0502	—	常に0	—	—	2
0503	ALCO	アラームコード		1	2
0504,0505	ALTO	アラーム発生時刻		2	4
1000 ~ 3FFF	PCMD	目標位置	○	2	4
	INP	位置決め幅	○	2	4
	VCMD	速度指令		2	4
(注)小さいポ ジションNo.から 順次割付けさ れています。	ZNMP	個別ゾーン境界+側	○	2	4
	ZNLP	個別ゾーン境界-側	○	2	4
	ACMD	加速度指令		1	2
	DCMD	減速度指令		1	2
	PPOW	押付け時電流制限値		1	2
	LPOW	負荷電流閾値		1	2
	CTLF	制御フラグ指定		1	2
8400,8401	TLMC	通算移動回数 ^(注1)		2	4
8402,8403	ODOM	通算走行距離 ^(注1)		2	4
841A,841B	TIMN	現在時刻(SCON-CA 専用)		2	4
8420,8421	TIMN	現在時刻(PCON-CA/CFA 専用)		2	4
842E,842F	TFAN	FAN 通算駆動時間(PCON-CFA 専用)		2	4
9000,9001	PNOW	現在位置モニタ	○	2	4
9002	ALMC	現在発生アラームコード照会		1	2
9003	DIPM	入力ポート照会		1	2
9004	DOPM	出力ポートモニタ照会		1	2
9005	DSS1	デバイスステータス1照会		1	2
9006	DSS2	デバイスステータス2照会		1	2
9007	DSSE	拡張デバイスステータス照会		1	2
9008,9009	STAT	システムステータス照会		2	4

アドレス[H]	記号	名 称	符号	レジスタサイズ*	バイト
900A,900B	VNOW	現在速度モニタ	○	2	4
900C,900D	CNOW	電流値モニタ	○	2	4
900E,900F	DEVI	偏差モニタ	○	2	4
9010,9011	STIM	システムタイマ照会		2	4
9012	SIPM	特殊入力ポート照会		1	2
9013	ZONS	ゾーンステータス照会		1	2
9014	POSS	位置決め完了ポジションNo.ステータス照会		1	2
9015	SSSE	拡張システムステータスレジスタ		1	2
901E	FBFC	カフィードバックデータモニタ	○	2	4

注1 PCON-CA/CFA、SCON-CA、ERC3専用

(3) クエリフォーマット

クエリメッセージでは、読取りを開始するレジスタのアドレスとレジスタの数を指定します。

1レジスタ=2バイト=16ビットデータ

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読出し
開始アドレス[H]	4	任意	(2)開始アドレス 一覧参照
レジスタの数[H]	4	任意	開始アドレス一覧参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(4) レスポンスフォーマット

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読出し
データバイト数[H]	2		クエリフォーマット内 指定レジスタ数×2
データ1[H]	4		
データ2[H]	4		
データ3[H]	4		
データ4[H]	4		
:	:		
:	:		
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	最大256		

(5) 使用例

下記に軸No.0 RCコントローラのアドレス9000_H～9009_Hまでを読み取るクエリ例を示します。

クエリ:010390000000A62[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '0'	39303030
レジスタの数[H]	'0', '0', '0', 'A'	30303041
エラーチェック[H]	'6', '2' (LRC計算による)	3632
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:010314000000000000B80162002000800031C7000800111C[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'1', '4' (20バイト= 10レジスタ)	3134
データ1[H]	'0', '0', '0', '0', '0', '0', '0', '0'	3030303030303030
データ2[H]	'0', '0', '0', '0'	30303030
データ3[H]	'B', '8', '0', '1'	42383031
データ4[H]	'6', '2', '0', '0'	36323030
データ5[H]	'2', '0', '0', '0'	32303030
データ6[H]	'8', '0', '0', '0'	38303030
データ7[H]	'3', '1', 'C', '7'	33314337
データ8[H]	'0', '0', '0', '8', '0', '0', '1', '1'	3030303830303131
エラーチェック[H]	'1', 'C' (LRC計算による)	3143
トレーラ	'CR', 'LF'	0D0A

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.2 アラーム詳細内容の読み取り《ALA0、ALC0、ALT0》

(1) 機能

最後に発生したアラームコード、アラーム詳細コードおよびアラーム発生時刻を読み取ります。アラームが発生していない場合は0_Hです。[詳細は、4.3.2(1)～(3)項を参照]

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ～10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘0’, ‘5’, ‘0’, ‘0’	アラーム詳細コード
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘6’	アドレス0500 _H ～0505 _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ～10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘C’	6レジスタ呼出=12バイト
データ1[H]	8	アラーム詳細コード アラームアドレス	アラーム詳細コード(0500 _H) [Hex] アラームアドレス(0501 _H) [Hex]
データ2[H]	8	アラームコード	アラームコード[Hex]
データ3[H]	8	アラーム発生時刻 ^(注1)	アラーム発生時刻[Hex]
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	35		

注1 RTC(カレンダー機能)搭載機種でRTC有効の場合①と、RTC無効、またはRTCが無い機種②ではデータの内容が異なります。

①アラーム発生時刻を示します。 ②電源投入後からの経過時間[msec]を示します。

(4) 使用例

軸No.0コントローラで最後に発生したアラーム内容(アドレス0500_H~0505_H)を読み取りした使用例を示します

クエリ:010305000006F1[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'0', '5', '0', '0'	30353030
レジスタの数[H]	'0', '0', '0', '6'	30303036
エラーチェック[H]	'F', '1' (LRC計算による)	4631
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:01030C0000FFFF000000E8172C643F24[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', 'C' (12バイト= 6レジスタ)	3043
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'F', 'F', 'F', 'F'	46464646
データ3[H]	'0', '0', '0', '0', '0', '0', 'E', '8'	3030303030304538
データ4[H]	'1', '7', '2', 'C', '6', '4', '3', 'F'	3137324336343346
エラーチェック[H]	'2', '4' (LRC計算による)	3234
トレーラ	'CR', 'LF'	0D0A

アラーム詳細コード:0000_H・・詳細コード無し

アラームアドレス:FFFF_H・・無効(詳細コード無し)

アラームコード:00E8_H=0E8・・エンコーダ AB 相断線エラー

アラーム発生時刻:172C643F_H(変換)⇒2012/04/26 19:53:35[変換は、4.3.2(4)項参照]

注1 レスポンス例のデータは一例であり、各種状態により異なります。

注2 アラームコードの詳細内容は、各コントローラの取扱説明書を参照ください。

6. 4. 3 ポジションデータの読み取り 《PCMD、INP、VCMD、ZNMP、ZNLP、ACMD、DCMD、PPOW、LPOW、CTLF》

(1) 機能

指定したポジションNo.に設定された値を読み取ります。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、ASCIIモード使用時256バイト中9バイト(ヘッダ+スレーブアドレス+ファンクションコード+エラーチェック+トレーラ)を除く247バイト分最大123レジスタ分(1レジスタ2バイト使用)のデータ照会が可能です。連続したアドレスの複数レジスタを一度の送受信で照会することも可能です。

アドレス [H]	各ポジションNo. の先頭アドレス [H]	先頭アド レスから のオフセット [H]	記号	レジスタ名称	符号	レジスタ サイズ*	バイト	単位
1000 ~ 3FFF	先頭アドレス= 1000 _H +(16 ×ポジションNo.)	+0	PCMD	目標位置	○	2	4	0.01mm
		+2	INP	位置決め幅	○	2	4	0.01mm
		+4	VCMD	速度指令		2	4	0.01mm/s
		+6	ZNMP	個別ゾーン境界 +側	○	2	4	0.01mm
		+8	ZNLP	個別ゾーン境界 -側	○	2	4	0.01mm
		+A	ACMD	加速度指令		1	2	0.01G
		+B	DCMD	減速度指令		1	2	0.01G
		+C	PPOW	押付け時電流 制限値		1	2	% (100%=FF _H)
		+D	LPOW	負荷電流閾値		1	2	% (100%=FF _H)
		+E	CTLF	制御フラグ指定		1	2	

ASCII
ファンクション
コード 03

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_{\text{H}} + (16 \times \text{ポジション No.})_{\text{H}} + \text{アドレス(オフセット値)}_{\text{H}}$$

例 ポジションNo.200 の速度指令レジスタを変更したい場合

$$1000_{\text{H}} + (16 \times 200 = 3200)_{\text{H}} + 4_{\text{H}}$$

$$= 1000_{\text{H}} + \text{C80}_{\text{H}} + 4_{\text{H}}$$

$$= 1\text{C84}_{\text{H}}$$

『1C84』がクエリ開始アドレス部入力値になります。

注 最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

(3) クエリフォーマット

クエリメッセージでは、読み取りを開始するレジスタのアドレスとレジスタのバイト数を指定します。

1レジスタ(1アドレス)=2バイト=16ビットデータ

フィールド名称	文字数 (バイト数)	ASCIIデータ 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0','3'	レジスタの読出し
開始アドレス[H]	4	任意	(2)開始アドレス一覧参照
レジスタの数[H]	4	任意	
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	CR/LF	
合計バイト数	17		

(4) レスポンスフォーマット

フィールド名称	文字数 (バイト数)	ASCIIデータ 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0','3'	レジスタの読出し
データバイト数[H]	2		クエリ指定レジスタの バイト数の合計
データ1[H]	4		
データ2[H]	4		
データ3[H]	4		
データ4[H]	4		
:	:		
:	:		
エラーチェック[H]	2	LRC計算結果	
トレーラ		CR/LF	
合計バイト数	最大256		

(5) 使用例

軸No.0コントローラのポジションNo.1(アドレス1010_H~1015_H)の目標位置、位置決め幅および速度指令を照会したクエリ例を示します。

クエリ: 010310100006D6[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0','1'	3031
ファンクションコード[H]	'0','3'	3033
開始アドレス[H]	'1','0','1','0'	31303130
レジスタの数[H]	'0','0','0','6' (6レジスタ)	30303036
エラーチェック[H]	'D','6' (LRC計算による)	4436
トレーラ	'CR','LF'	0D0A

レスポンスは以下ようになります。

●レスポンス(前後にサイレントインターバルが入ります)

01030C000007D000001F4000003A98E8[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0','1'	3031
ファンクションコード[H]	'0','3'	3033
データバイト数[H]	'0','C' (12バイト= 6レジスタ)	3043
データ1[H]	'0','0','0','0','0','7','D','0' (目標位置照会)	3030303030374430
データ2[H]	'0','0','0','0','1','F','4','0' (位置決め幅照会)	3030303031463430
データ3[H]	'0','0','0','0','3','A','9','8' (速度指令照会)	3030303033413938
エラーチェック[H]	'E','8' (LRC計算による)	4538
トレーラ	'CR','LF'	0D0A

ASCII
ファンクション
コード 03

目標位置は、7D0_H→10進数に変換→2000×[単位0.01mm]= 20.00[mm]

位置決め幅は、1F40_H→10進数に変換→8000×[単位0.01mm]=80.00[mm]

速度指令は、3A98_H→10進数に変換→15000×[単位0.01mm]=150.00[mm]

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.4 通算移動回数の読み取り《TLMC》

(1) 機能

通算移動回数を読み取ります。[詳細は、4.3.2(8)項を参照]

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘8’, ‘4’, ‘0’, ‘0’	通算移動回数
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス8400 _H ~8401 _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ1[H]	4	通算移動回数	通算移動回数[Hex](上位)
データ2[H]	4	通算移動回数	通算移動回数[Hex](下位)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの通算移動回数(アドレス8400_H~8401_H)を読み取りした使用例を示します。

クエリ:01038400000276[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'8', '4', '0', '0'	38343030
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'7', '6' (LRC計算による)	3736
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:0103040000021FD7[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4'	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'0', '2', '1', 'F'	30323146
エラーチェック[H]	'D', '7' (LRC計算による)	4337
トレーラ	'CR', 'LF'	0D0A

通算移動回数は、21F_H→10進数に変換→543[回]

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.5 通算走行距離の読み取り《ODOM》(1m 単位)

(1) 機能

通算走行距離を 1m 単位で読み取ります。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘8’, ‘4’, ‘0’, ‘2’	通算走行距離
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス8402 _H ~8403 _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ1[H]	4	通算走行距離	通算走行距離[Hex](上位)
データ2[H]	4	通算走行距離	通算走行距離[Hex](下位)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの通算走行距離(アドレス8402_H~8403_H)を読み取りした使用例を示します
 ケーリ:01038402000274[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'8', '4', '0', '2'	38343032
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'7', '4' (LRC計算による)	3734
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:0103040000409E1A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4'	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'4', '0', '9', 'E'	34303945
エラーチェック[H]	'1', 'A' (LRC計算による)	3141
トレーラ	'CR', 'LF'	0D0A

通算走行距離は 0000409E_H→10進数に変換→16542m

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.6 現在時刻の読み取り《TIMN》

(1) 機能

現在時刻を読み取ります。

[PCON-CA/CFA および SCON-CA 専用]

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	SCON-CA : ‘8’, ‘4’, ‘1’, ‘A’ PCON-CA/CFA : ‘8’, ‘4’, ‘2’, ‘0’	現在時刻モニタ
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス8402 _H ~8403 _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ[H]	8	現在時刻	時刻への変換は(4)参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 読み取ったデータを時刻に変換

読み取ったデータは、コントローラの設定により、現在の時刻または時間を出力します。

- ①RTC(カレンダー機能)を有効に設定している場合、現在時刻になります。
- ②RTCを無効に設定している場合、コントローラ電源投入を基準とした経過時間[sec]になります。

①現在時刻の計算方法

読み取った現在時刻のデータは、基準時刻(2000年1月1日00時00分00秒)からの経過秒を示しています。

基準時刻からの経過秒をS、経過分をM、経過時をH、経過日をD、経過年をYとし、次の式で計算を行います。

$$\begin{aligned}
 S &= \text{読み込んだ現在時刻のデータ} \\
 M &= S/60(\text{小数点以下切捨て}) \\
 H &= M/60(\text{小数点以下切捨て}) \\
 D &= H/24(\text{小数点以下切捨て}) \\
 Y &= D/365.25(\text{小数点以下切捨て}) \\
 L(\text{閏年計算}) &= Y/4(\text{小数点切り上げ})
 \end{aligned}$$

時刻の秒をSA、分をMA、時をHA、今年になってからの経過日をDA、年をYAとすると、以下の式で時刻を計算できます。

$$\begin{aligned}
 SA &= S/60 \text{ の余り} \\
 MA &= M/60 \text{ の余り} \\
 HA &= H/24 \text{ の余り} \\
 DA &= D - (Y \times 365 + L) \dots DA \text{ から一月ごとの日数を減算することで月日を求めます。} \\
 YA &= Y + 2000 \text{ (西暦)}
 \end{aligned}$$

例) 現在時刻のデータが172C1B8B_Hだった場合

【手順1】10進数に変換: $S = 172C1B8B_H \Rightarrow 388766603$

【手順2】M、H、D、Y、Lを計算します。

$$\begin{aligned}
 M &= 388766603/60 = 6479443 \\
 H &= 6479443/60 = 107990 \\
 D &= 107990/24 = 4499 \\
 Y &= 4499/365.25 = 12 \\
 L &= 12/4 = 3
 \end{aligned}$$

【手順3】SA、MA、HA、およびDAを求めます。

$$\begin{aligned}
 SA &= 388766603/60 \text{ の余り} = 23 \\
 MA &= 6479443/60 \text{ の余り} = 43 \\
 HA &= 107990/24 \text{ の余り} = 14 \\
 DA &= 4499 - (12 \times 365 + 3) \\
 &= 116(\text{今年になって116日経過し、現在は117日目}) \\
 \text{月日} &= 117 - \{31(1\text{月}) - 29(2\text{月}) - 31(3\text{月})\} \\
 &= 26(4\text{月分を減算すると負数になってしまうので、発生時は4月26日}) \\
 YA &= 12 + 2000 = 2012 \\
 &\text{以上より、現在時刻は、2012年4月26日14時43分23秒となります。}
 \end{aligned}$$

②経過時間の計算方法

例) 現在時刻のデータが E1B8B_H だった場合

10 進数に変換: E1B8B_H ⇒ 924555

したがって、電源投入後 924555 sec(256 時間 49 分 15 秒)経過となります。

(5) 使用例

軸No.0のPCON-CAの現在時刻(アドレス8420_H~8421_H)を読み取りした使用例を示します。

クエリ:01038420000256[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'8', '4', '2', '0'	38343230
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'5', '6' (LRC計算による)	3536
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:010304172C1B8B56[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4'	3034
データ[H]	'1', '7', '2', 'C', '1', 'B', '8', 'B'	3137324331423842
エラーチェック[H]	'5', '6' (LRC計算による)	3536
トレーラ	'CR', 'LF'	0D0A

現在時刻は、2012 年 4 月 26 日 14 時 43 分 23 秒

注 レスポンス例のデータは一例であり、各種状態により異なります。



6.4.7 ファン通算駆動時間の読み取り《TFAN》

(1) 機能

ファンの通算駆動時間を読み取ります。(1秒単位)

[PCON-CFA 専用]

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘8’, ‘4’, ‘2’, ‘E’	ファン通算駆動時間
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス842E _H ~842F _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ1[H]	4	ファン通算駆動時間	ファン通算駆動時間[Hex](上位)
データ2[H]	4	ファン通算駆動時間	ファン通算駆動時間[Hex](下位)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラのファン通算駆動時間(アドレス842E_H~842F_H)を読み取りした使用例を示します。

クエリ:0103842E000248[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'8', '4', '2', 'E'	38343245
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'4', '8' (LRC計算による)	3438
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:010304000002AF47

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4'	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'0', '2', 'A', 'F'	30324146
エラーチェック[H]	'4', '7' (LRC計算による)	3437
トレーラ	'CR', 'LF'	0D0A

ファン通算駆動時間は 000002AF_H→10進数に変換→687[秒]

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.8 現在位置の読み取り(0.01mm 単位)モニタ 《PNOW》

(1) 機能

現在位置を0.01mm単位で読み取ります。符号は有効です。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '0'	現在位置モニタ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス9000 _H ~9001 _H 呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ1[H]	4	現在値による	現在値データ(Hex)
データ2[H]	4	現在値による	現在値データ(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例(軸No.が0の場合)

下記に軸No.0 RCコントローラのアドレス9000_Hを読み取るクエリ例を示します。

クエリ: 0103900000026A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス	'0', '1'	3031
ファンクションコード	'0', '3'	3033
開始アドレス	'9', '0', '0', '0'	39303030
レジスタの数	'0', '0', '0', '2'	30303032
エラーチェック	'6', 'A'	3641
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス: 010304000013885D[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'1', '3', '8', '8'	31333838
エラーチェック[H]	'5', 'D' (LRC計算による)	3544
トレーラ	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

現在位置は『00001388』→10進数に変換→5000(×0.01mm)

現在位置は 50mm

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.9 現在発生アラームコードの読み取り《ALMC》

(1) 機能

コントローラの正常状態、またはアラーム状態(コールトスタートレベル、動作解除レベルおよびメッセージレベル)を示すコードを読み取ります。

[コードの詳細内容は各コントローラの取扱説明書参照]

(2) クエリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '2'	現在発生アラームコード
レジスタの数[H]	4	'0', '0', '0', '1'	アドレス9002 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '2'	1レジスタ呼出=2バイト
データ1[H]	4	アラームコード	アラームコード(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

下記に軸No.0 RCコントローラのアドレス9002_Hまでを読み取りしたクエリ例を示します。

クエリ: 01039002000169[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス	'0', '1'	3031
ファンクションコード	'0', '3'	3033
開始アドレス	'9', '0', '0', '2'	39303032
レジスタの数	'0', '0', '0', '1'	30303031
エラーチェック	'6', '9'	3639
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス: 01030200E812[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
ヘッダ	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'0', '0', 'E', '8'	30304538
エラーチェック[H]	'1', '2' (LRC計算による)	3132
トレーラ	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

このとき発生している最重要アラームは 『0E8』_H

A,B 相断線検出アラームです。

[コードの詳細内容は各コントローラの取扱説明書参照]

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.10 I/Oポート入力信号の状態読み取り《DIPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート入力値を読み取ります。

データは RCコントローラが入力として認識しているポートの状態です。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '3'	入力ポートモニタレジスタ
レジスタの数[H]	4	'0', '0', '0', '1'	アドレス9003 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、17アドレス当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '2'	1レジスタ呼出=2バイト
データ1[H]	4	DI入力値	DI入力値(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

軸No.0コントローラの入力ポート(アドレス9003_H)を読み取りした使用例を示します

クエリ:01 03 90 03 00 01 68 [CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '3'	39303033
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '8' (LRC計算による)	3638
エンド	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:01 03 02 B8 01 14 [CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'B', '8', '0', '1'	42383031
エラーチェック[H]	'1', '4' (LRC計算による)	3134
エンド	'CR', 'LF'	0D0A

入力ポートデータ部は『B801』_H →2進数変換『1011100000000001』

注 レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け [詳細は、各 RC コントローラの取扱説明書を参照]

各 RC コントローラの PIO パターンごとのポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)				ERC3(PIOタイプ)		
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン0	PIOパターン1	PIOパターン2
IN0	PC1	ST0	PC1	PC1	PC1	ST0	PC1
IN1	PC2	ST1	PC2	PC2	PC2	ST1	PC2
IN2	PC4	ST2	PC4	PC4	PC4	ST2	PC4
IN3	HOME	0	PC8	PC8	HOME	0	PC8
IN4	CSTR	RES	CSTR	CSTR	CSTR	RES	CSTR
IN5	*STP	*STP	*STP	*STP	*STP	*STP	*STP
IN6	0	0	0	0	0	0	0
IN7	0	0	0	0	0	0	0
IN8	0	0	0	0	0	0	0
IN9	0	0	0	0	0	0	0
IN10	0	0	0	0	0	0	0
IN11	0	0	0	0	0	0	0
IN12	0	0	0	0	0	0	0
IN13	0	0	0	0	0	0	0
IN14	0	0	0	0	0	0	0
IN15	0	0	0	0	0	0	0

ポート	PCON-C/CG/CF/CA/CFA						PCON-CA/CFA	PCON-PL/PO	
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン4	PIOパターン5	PIOパターン6 (パルス列モード)	PIOパターン0	PIOパターン1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	RES	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	TL	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	CSTR	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	DCLR	0	0
IN6	0	MODE	PC64	PC64	ST6	0	BKRL	0	0
IN7	0	JISL	PC128	PC128	0	0	RMOD	0	0
IN8	0	JOG+	0	PC256	0	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0	0

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン0	PIO パターン1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

ポート	SCON-C/CA						SCON-CA		SCON-C/CA
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン6	PIO パターン7	PIOパターン0 (パルス列モード)
IN0	PC1	PC1	PC1	PC1	ST0	ST0	PC1	ST0	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	PC2	ST1	RES
IN2	PC4	PC4	PC4	PC4	ST2	ST2	PC4	ST2	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	PC8	ST3	TL
IN4	PC16	PC16	PC16	PC16	ST4	0	PC16	ST4	CSTR
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0	DCLR
IN6	0	MODE	PC64	PC64	ST6	0	0	0	BKRL
IN7	0	JISL	PC128	PC128	0	0	0	0	RMOD
IN8	0	JOG+	0	PC256	0	0	CLBR	CLBR	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	BKRL	BKRL	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0
IN11	HOME	HOME	HOME	HOME	HOME	0	HOME	HOME	0
IN12	*STP	*STP	*STP	*STP	*STP	0	*STP	*STP	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	CSTR	0	0
IN14	RES	RES	RES	RES	RES	RES	RES	RES	0
IN15	SON	SON	SON	SON	SON	SON	SON	SON	0

6.4.11 I/Oポート出力信号の状態読み取り《DOPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート出力値を読み取ります。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '4'	出力ポートモニタレジスタ
レジスタの数[H]	4	'0', '0', '0', '1'	アドレス9004 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '2'	1レジスタ呼出=2バイト
データ1[H]	4	DO出力値	DI出力値(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

軸No.0コントローラの入力ポート(アドレス9004_H)を読み取りした使用例を示します

クエリ:01039004000167[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '4'	39303034
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '7' (LRC計算による)	3637
エンド	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:010302740086[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'7', '4', '0', '0'	37343030
エラーチェック[H]	'8', '6' (LRC計算による)	3836
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

出力ポート部は『7400』_H →2進数変換『0111010000000000』

注 レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け [詳細は、各 RC コントローラの取扱説明書を参照]

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)				ERC3(PIOタイプ)		
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン0	PIOパターン1	PIOパターン2
OUT0	PEND	PE0	PEND	PEND	PEND	PE0	PEND
OUT1	HEND	PE1	HEND	HEND	HEND	PE1	HEND
OUT2	ZONE	PE2	ZONE	ZONE	ZONE1	PE2	PZONE/ ZONE1
OUT3	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM
OUT4	0	0	0	0	0	0	0
OUT5	0	0	0	0	0	0	0
OUT6	0	0	0	0	0	0	0
OUT7	0	0	0	0	0	0	0
OUT8	0	0	0	0	0	0	0
OUT9	0	0	0	0	0	0	0
OUT10	0	0	0	0	0	0	0
OUT11	0	0	0	0	0	0	0
OUT12	0	0	0	0	0	0	0
OUT13	0	0	0	0	0	0	0
OUT14	0	0	0	0	0	0	0
OUT15	0	0	0	0	0	0	0

ポート	PCON-C/CG/CF/CA/CFA						PCON-CA/CFA	PCON-PL/PO	
	PIOパターン0	PIOパターン1	PIOパターン2	PIOパターン3	PIOパターン4	PIOパターン5	PIOパターン6 (パルス列モード)	PIOパターン0	PIOパターン1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PWR	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	SV	INP	INP/TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	INP	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	HEND	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	TLR	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	*ALM	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	*EMGS	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	RMDS	0	0
OUT8	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1 ^(注)	PZONE/ ZONE1 ^(注)	PM256	PZONE/ ZONE2 ^(注)	PZONE/ ZONE2 ^(注)	ALM1	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	ALM4	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	ALM8	0	0
OUT12	SV	SV	SV	SV	SV	SV	*ALML ^(注)	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	ZONE1	0	0
OUT15	LOAD/ TRQS/ ALML ^(注)	ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	LOAD/ TRQS/ ALML ^(注)	ALML ^(注)	ZONE2	0	0

注 CA/CFA タイプ限定

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン0	PIO パターン1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	TRQS	0	TRQS	TRQS	TRQS	0	0	0

ポート	SCON-C/CA						SCON-CA		SCON-C/CA
	PIO パターン0	PIO パターン1	PIO パターン2	PIO パターン3	PIO パターン4	PIO パターン5	PIO パターン6	PIO パターン7	PIOパターン0 (パルス列ポート)
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PM1	PE0	PWR
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	PM2	PE1	SV
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	PM4	PE2	INP
OUT3	PM8	PM8	PM8	PM8	PE3	0	PM8	PE3	HEND
OUT4	PM16	PM16	PM16	PM16	PE4	0	PM16	PE4	TLR
OUT5	PM32	PM32	PM32	PM32	PE5	0	TRQS	TRQS	*ALM
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	LOAD	LOAD	*EMGS
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	CEND	CEND	RMDS
OUT8	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1 ^(注)	PZONE/ ZONE1 ^(注)	PM256	PZONE/ ZONE2 ^(注)	PZONE/ ZONE2 ^(注)	PZONE/ ZONE1	PZONE/ ZONE1	ALM1
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	HEND	HEND	ALM4
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	PEND	PEND	ALM8
OUT12	SV	SV	SV	SV	SV	SV	SV	SV	*OVLW ^(注) / *ALML ^(注)
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	ZONE1
OUT15	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	ZONE2

注 CA タイプ限定

6.4.12 コントローラ状態信号の読み取り1《DSS1》

(1) 機能

コントローラ内のステータスを読み取ります。

[4.3.2 (12) デバイスステータスレジスタ1 内容参照]

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘0’, ‘5’	デバイスステータス レジスタ1
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9005 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1アドレス当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	ステータス1	ステータス1(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9005_H)を読み取る使用例を示します

クエリ:01 03 90 05 00 01 66[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '5'	39303035
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '6' (LRC計算による)	3636
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 30 88 42[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'3', '0', '8', '8'	33303838
エラーチェック[H]	'4', '2' (LRC計算による)	3432
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.13 コントローラ状態信号の読み取り2 《DSS2》

(1) 機能

コントローラ内のステータス2を読み取ります。

[4.3.2 (13) デバイスステータスレジスタ2 内容参照]

(2) クエリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘0’, ‘6’	デバイスステータス レジスタ2
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9006 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	コントローラ内のステータス
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	ステータス2	ステータス2(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのデバイスステータス2(アドレス9006_H)を読み取る使用例を示します

クエリ:01 03 90 06 00 01 65[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '6'	39303036
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '5' (LRC計算による)	3635
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 80 00 7A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'8', '0', '0', '0'	38303030
エラーチェック[H]	'7', 'A' (LRC計算による)	3741
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.14 コントローラ状態信号の読み取り3 《DSSE》

(1) 機能

コントローラ内のステータス(拡張デバイス)を読み取ります。

[4.3.2 (14) 拡張デバイスステータスレジスタ内容参照]

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘0’, ‘7’	拡張デバイスステータス レジスタ
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9007 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	拡張ステータス	拡張ステータス(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの拡張デバイスステータス(アドレス9007_H)を読み取る使用例を示します

クエリ:01 03 90 07 00 01 64[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '7'	39303037
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '4' (LRC計算による)	3634
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 33 C7 00[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'3', '3', 'C', '7'	33334337
エラーチェック[H]	'0', '0' (LRC計算による)	3030
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.15 コントローラ状態信号の読み取り4《STAT》

(1) 機能

コントローラの内部動作状態を読み取ります。

[4.3.2 (15) システムステータスレジスタ内容参照]

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '8'	システムステータス レジスタ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス9008 _H ~9009 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	コントローラ内のステータス
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ[H]	8	システムステータス	システムステータス(Hex)
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラのシステムステータス(アドレス9008_H)を読み取る使用例を示します

クエリ:01 03 90 08 00 02 62[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', '8'	39303038
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'6', '2' (LRC計算による)	3632
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 0C 00 11 DB[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '0', '0', 'C'	30303043
データ2[H]	'0', '0', '1', '1'	30303131
エラーチェック[H]	'D', 'B' (LRC計算による)	4442
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.16 現在速度の読み取り《VNOW》

(1) 機能

モータ実速度を読み取ります。移動方向により±に変化します。

単位は0.01mm/sです。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘0’, ‘A’	現在速度モニタ
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス900A _H ~900B _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ[H]	8	現在速度	現在速度(Hex) 単位は0.01mm/s
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの速度(アドレス900A_H)を読み取る使用例を示します

クエリ:01 03 90 0A 00 02 60[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', 'A'	39303041
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'6', '0' (LRC計算による)	3630
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 26 FC D6[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'2', '6', 'F', 'C'	32364643
エラーチェック[H]	'D', '6' (LRC計算による)	4436
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

現在速度は『000026FC』→10進数に変換→9980(×0.01mm/s)

現在速度モジュールは 99.8mm/s

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.17 電流値の読み取り《CNOW》

(1) 機能

モータ電流のモニターデータです。

単位はmAです。

トルク電流指令値が格納されます。

(2) クエリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', '0'	電流値モニタ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス900C _H ~900D _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ[H]	8	モータ電流の モニタ	モータ電流のモニタ(Hex) 単位はmA
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの電流値(アドレス900C_H)を読み取る使用例を示します

クエリ:01 03 90 0C 00 02 5E[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', 'C'	39303043
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'5', 'E' (LRC計算による)	3545
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 01 C8 2F[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'0', '1', 'C', '8'	30314338
エラーチェック[H]	'2', 'F' (LRC計算による)	3246
エンド	'CR', 'LF'	0D0A

モータ電流値は『000001C8』→10進数に変換→456(mA)

電流モニタ値は 456mA

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.18 偏差の読み取り《DEVI》

(1) 機能

1ms周期毎の位置指令値とフィードバック値(実位置)の偏差量を読み取ります。単位はpulseです。

モータ機械角1回転あたりのパルス数は使用エンコーダにより異なります。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '0', 'E'	偏差モニタ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス900E _H ~900F _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ[H]	8	偏差モニタ	偏差モニタ(Hex) 単位はpulse
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの偏差量(アドレス90E_H)を読み取る使用例を示します

クエリ:01 03 90 0E 00 02 5C[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '0', 'E'	39303045
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'5', 'C' (LRC計算による)	3543
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 00 83 75[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'0', '0', '8', '3'	30303833
エラーチェック[H]	'7', '5' (LRC計算による)	3735
エンド	'CR', 'LF'	0D0A

偏差モニタは『00000083』→10進数に変換→131pulse

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量は 131pulse

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.19 電源投入後の積算時間の読み取り《STIM》

(1) 機能

コントローラ電源投入時からの積算時間を読み取ります。

単位はmsです。

ソフトウェアリセットではクリアされません。

(2) クエリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '1', '0'	システムタイマ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス9010 _H ~9011 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ[H]	8	システムタイマ	システムタイマ(Hex) 単位はms
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの起動時間(アドレス9010_H)を読み取り使用例を示します

クエリ:01 03 90 10 00 02 5A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', '0'	39303130
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'5', 'A' (LRC計算による)	3541
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 02 38 C0 94 6A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (4バイト= 2レジスタ)	3034
データ1[H]	'0', '2', '3', '8'	30323338
データ2[H]	'C', '0', '9', '4'	43303934
エラーチェック[H]	'6', 'A' (LRC計算による)	3641
エンド	'CR', 'LF'	0D0A

システムタイムは『0238C094』→10進数に変換→37273748ms

コントローラ電源投入時からの積算時間は 10.3538 時間

注 レスポンス例のデータは一例であり、各種状態により異なります。

6. 4. 20 特殊入力ポートの入力信号状態の読み取り《SIPM》

(1) 機能

通常の入力ポート以外の入力ポートの状態を読み取ります。

[特殊入力ポートの内容は「4. 3. 2 (16) 特殊ポートモニタレジスタ内容参照」]

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘1’, ‘2’	特殊ポートモニタ
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9012 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	特殊ポートモニタ	4.3.2(16) 一覧表参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの特異入力ポート(アドレス9012_H)を読み取る使用例を示します

クエリ:01 03 90 12 00 01 59[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', '2'	39303132
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'5', '9' (LRC計算による)	3539
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 03 00 F7[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'0', '3', '0', '0'	30333030
エラーチェック[H]	'F', '7' (LRC計算による)	4637
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.21 ゾーン出力信号の状態の読み取り《ZONS》

(1) 機能機能

ゾーン出力の状態を読み取ります。

[4.3.2 (17) ゾーンステータスレジスタ内容参照]

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘1’, ‘3’	ゾーンステータス照会
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9013 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	ゾーンステータス	4.3.2(17)一覧表参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのゾーンステータス(アドレス9013_H)を読み取る使用例を示します

クエリ:01 03 90 13 00 01 58[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', '3'	39303133
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'5', '8' (LRC計算による)	3538
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 00 00 FA[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'0', '0', '0', '0'	30303030
エラーチェック[H]	'F', 'A' (LRC計算による)	4641
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

注 レスポンス例のデータは一例であり、各種状態により異なります。

6. 4. 22 位置決め完了ポジションNo.照会 《POSS》

(1) 機能

完了ポジション番号を読み取ります。

[4. 3. 2 (18) ポジション番号ステータスレジスタ内容参照]

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘1’, ‘4’	ポジション番号ステータス
レジスタの数[H]	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9014 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	ポジション番号 ステータス	4.3.2(18)一覧表参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの位置決め完了ポジション(アドレス9014_H)を読み取る使用例を示します

クエリ:01 03 90 14 00 01 57[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', '4'	39303134
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'5', '7' (LRC計算による)	3537
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 00 00 FA[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2' (2バイト= 1レジスタ)	3032
データ1[H]	'0', '0', '0', '0'	30303030
エラーチェック[H]	'F', 'A' (LRC計算による)	4641
エンド	'CR', 'LF'	0D0A

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.23 コントローラ状態信号の読み取り5 《SSSE》

(1) 機能

コントローラの内部動作状態を読み取ります。

[4.3.2 (19) 拡張システムステータスレジスタ内容を参照]

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス[H]	4	‘9’, ‘0’, ‘1’, ‘5’	拡張システムステータス レジスタ
レジスタの数[H]	1	‘0’, ‘0’, ‘0’, ‘1’	アドレス9015 _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	14		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	‘0’, ‘3’	コントローラ内のステータス
データバイト数[H]	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ[H]	4	拡張システムステータス	拡張システムステータス[Hex]
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの拡張システムステータス(アドレス9015_H)を読み取りした使用例を示します。

クエリ:01039015000156[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', '5'	39303135
レジスタの数[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'5', '6' (LRC計算による)	3536
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:0103020100F9[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '2'	3032
データ1[H]	'0', '1', '0', '0'	30313030
エラーチェック[H]	'F', '9' (LRC計算による)	4639
エンド	'CR', 'LF'	0D0A

注 レスポンス例のデータは一例であり、各種状態により異なります。

6.4.24 カフィードバックデータの読み取り《FBFC》・SCON-CA 専用

(1) 機能

ロードセル測定値(押付け力)のモニタデータを読み取ります。

単位は0.01Nです。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
開始アドレス[H]	4	'9', '0', '1', 'E'	カフィードバックデータモニタ
レジスタの数[H]	4	'0', '0', '0', '2'	アドレス901E _H ~901F _H の呼出
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '3'	レジスタの読み出し
データバイト数[H]	2	'0', '4'	2レジスタ呼出=4バイト
データ[H]	8	ポジション番号 ステータス	現在の押付け力[N] 単位は0.01N
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラに接続されているロードセルの現在測定値を読み取りした使用例を示します。

クエリ:01 03 90 0A 00 02 4C[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
開始アドレス[H]	'9', '0', '1', 'E'	39393145
レジスタの数[H]	'0', '0', '0', '2'	30303032
エラーチェック[H]	'4', 'C' (LRC計算による)	3443
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 03 E4 11[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '3'	3033
データバイト数[H]	'0', '4' (2バイト= 1レジスタ)	3034
データ1[H]	'0', '0', '0', '0'	30303030
データ2[H]	'0', '3', 'E', '4'	30334534
エラーチェック[H]	'1', '1' (LRC計算による)	3131
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 03

例 1) 現在のロードセル測定値は『000003E4』→10進数に変換→996(×0.01N)→
現在の押付け力は 9.96N

例 2) 現在のロードセル測定値が『FFFFFF35』と読み取れた時、(引張り状態^(注2))→
FFFFFFFF_H−FFFFFF35_H+1(必ず1を加算)→
10進数に変換→203(×0.01N)→
現在の引張り力^(注2)は 2.03N

注 1 レスポンス例のデータは一例であり、各種状態により異なります。

注 2 引張り方向に力を加えるとロードセルを破損します。

6.5 動作指令および、データ書き換え(使用ファンクションコード 05)

6.5.1 コイルへの書込み

※) [6.2 ASCIIコード表](#)
を参照ください。

(1) 機能

スレーブのDO(Discrete Output)の状態をON/OFFのいずれかに変更(書込み)します。プロトキャストの場合には、全スレーブの同じアドレスのコイルを書換えます。

(2) 開始アドレス一覧

開始アドレス[H]	記号	機能名称
0401	SFTY	セーフティ速度指令
0403	SON	サーボ ON 指令
0407	ALRS	アラームリセット指令
0408	BKRL	ブレーキ強制解除指令
040A	STP	一時停止指令
040B	HOME	原点復帰指令
040C	CSTR	位置決め動作起動指令
0411	JISL	ジョグ/インテグ切替え
0414	MOD	ティーチモード指令
0415	TEAC	ポジションデータ取込み指令
0416	JOG+	ジョグ+指令
0417	JOG-	ジョグ-指令
0418	ST7	スタートポジション 7(電磁弁モード)
0419	ST6	スタートポジション 6(電磁弁モード)
041A	ST5	スタートポジション 5(電磁弁モード)
041B	ST4	スタートポジション 4(電磁弁モード)
041C	ST3	スタートポジション 3(電磁弁モード)
041D	ST2	スタートポジション 2(電磁弁モード)
041E	ST1	スタートポジション 1(電磁弁モード)
041F	ST0	スタートポジション 0(電磁弁モード)
0426	CLBR	ロードセルキャリブレーション指令
0427	PMSL	PIO/Modbus 切替え指定
042C	STOP	減速停止

6.5.2 セーフティ速度有効/無効切替《SFTY》

(1) 機能

ユーザパラメータNo.35の「セーフティ速度」で指定された速度の有効/無効切替を行います。
MANUモード時に有効にしますと、全ての移動指令速度が制限されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,DOへの1点書き込み
開始アドレス[H]	4	'0', '4', '0', '1'	セーフティ速度指令
変更データ[H]	4	任意	セーフティ速度有効: 'F', 'F', '0', '0' セーフティ速度無効: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。
不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのセーフティ速度を有効にする使用例を示します。

クエリ:01 05 04 01 FF 00 F6

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', '1'	30343031
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'F', '6' (LRC計算による)	4636
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.3 サーボ ON/OFF 《SON》

(1) 機能

サーボ ON/OFF の制御を行います。

変更データ部をサーボ ON 状態にするとメカパラメータの「サーボオン遅延時間」経過後にサーボ ON 状態へと遷移します。ただし以下の条件を満たしている必要があります。

- ・デバイスステータスレジスタ 1 の EMG ステータスビットが 0
- ・デバイスステータスレジスタ 1 の重故障ステータスビットが 0
- ・デバイスステータスレジスタ 2 のイネーブルステータスビットが 1
- ・システムステータスレジスタの自動サーボ OFF 中ステータスビットが 0

(2) クエリフォーマット

フィールド名称	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル, D0への1点書き込み
開始アドレス[H]	4	'0', '4', '0', '3'	サーボON/OFF指令
変更データ[H]	4	任意	サーボON: 'F', 'F', '0', '0' サーボOFF: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

注 上位との通信前に、ティーチングボックス、パソコンソフトを接続し、サーボ OFF 動作をした後に、非接続にした場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIO ポートの接続を外す際はサーボ ON の状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのサーボをONする使用例を示します

クエリ:01 05 04 03 FF 00 F4

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', '3'	30343033
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'F', '4' (LRC計算による)	4634
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.4 アラームリセット《ALRS》

(1) 機能

アラームリセットのエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、アラームリセットを行います。

ただしアラーム要因が解消されていないと再びアラームとなります。

また一時停止中にアラームリセットのエッジを立てると残移動量のキャンセルが行われます。

アラームリセットを行ったら 必ず変更データを0000_Hにして書き込みを行い、通常の状態に戻してください。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,D0への1点書き込み
開始アドレス[H]	4	'0', '4', '0', '7'	アラームリセット指令
変更データ[H]	4	任意	アラームリセット実行: 'F', 'F', '0', '0' 通常: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるかもしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのアラームをリセットする使用例を示します

- 1回目 01 05 04 07 FF 00 F0 ...アラームリセット実行
 2回目 01 05 04 07 00 00 EF ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	‘:’	3A
スレーブアドレス[H]	‘0’, ‘1’	3031
ファンクションコード[H]	‘0’, ‘5’	3035
開始アドレス[H]	‘0’, ‘4’, ‘0’, ‘7’	30343037
変更データ[H]	1回目: ‘F’, ‘F’, ‘0’, ‘0’ 2回目: ‘0’, ‘0’, ‘0’, ‘0’ (アラームリセット終了後に0000 _H を書き込んで 通常状態に戻してください)	46463030 30303030
エラーチェック[H]	1回目: ‘F’, ‘0’ (LRC計算による) 2回目: ‘E’, ‘F’ (LRC計算による)	4630 4546
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.5 ブレーキ強制解除《BKRL》

(1) 機能

ブレーキの制御はサーボON/OFFと連動して行われますが、ブレーキがONの状態でも強制的に解除することができます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘0’, ‘8’	ブレーキ強制解除指令
変更データ[H]	4	任意	ブレーキ強制解除: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

注 上位との通信前に、パソコン対応ソフトなどのティーチングツールを接続し、サーボ OFF 動作をした後に、接続を外した場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボ ON の状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのブレーキを解除する使用例を示します

クエリ:01 05 04 08 FF 00 EF

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', '8'	30343038
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', 'F' (LRC計算による)	4546
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.6 一時停止《STP》

(1) 機能

移動中に一時停止指令を行うと減速停止を行い、再び通常状態にセットされると残移動量の移動を再開します。

一時停止指令の状態では、モータの移動は全て禁止されます。

一時停止指令中にアラームリセット指令ビットが立てられた場合は残移動量がキャンセルされます。

一時停止指令の入力が原点復帰動作中で、押付け反転前ならば移動指令が保留され、押付け反転後では原点復帰を最初からやり直します。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード[H]	2	'0', '5'	コイル,DOへの1点書き込み
開始アドレス[H]	4	'0', '4', '0', 'A'	一次停止指令
変更データ[H]	4	任意	一時停止指令: 'F', 'F', '0', '0' 通常: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを一時停止する使用例を示します

クエリ:01 05 04 0A FF 00 ED

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', 'A'	30343041
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', 'D' (LRC計算による)	4544
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.7 原点復帰《HOME》

(1) 機能

原点復帰指令のエッジを立てる(変更データが0000_Hの状態にFF00_Hを書き込む)と、原点復帰動作を行います。原点復帰が完了するとHENDビットが1になります。

原点復帰指令は、原点復帰が完了していても入力可能です。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,DOへの1点書き込み
開始アドレス[H]	4	'0', '4', '0', 'B'	原点復帰指令
変更データ[H]	4	任意	原点復帰実行: 'F', 'F', '0', '0' 通常: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

注 上位との通信前に、パソコン対応ソフトなどのティーチングツールを接続し、サーボ OFF 動作をした後に、接続を外した場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0 コントローラを原点復帰する使用例を示します

クエリ

1回目 01 05 04 0B 00 00 EB ...通常状態に設定

2回目 01 05 04 0B FF 00 EC ...原点復帰実行

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', 'B'	30343042
変更データ[H]	1回目: '0', '0', '0', '0' 2回目: 'F', 'F', '0', '0' (エッジを立てる為に2回データを送信してください。)	30303030 46463030
エラーチェック[H]	1回目: 'E', 'B' (LRC計算による) 2回目: 'E', 'C' (LRC計算による)	4542 4543
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

ASCII
ファンクション
コード 05

6.5.8 位置決め動作起動指令《CSTR》

(1) 機能

位置決め動作起動指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、ポジション番号指定レジスタ(POSR:0D03_H)内のポジション番号の指定位置に移動します。ポジションスタート指令状態のまま(FF00_Hを書き込んだまま)ですと位置決め幅内に入っても完了ポジションは出力されません。(0000_Hを書き込んで通常状態に戻してください。)

電源投入後、一度も原点復帰動作を行っていない状態(HENDビットが0の状態)では、原点復帰動作を実行した後に目標位置に移動を開始します。

※目標位置および速度等の動作パラメータは、全てコントローラ内部のポジションテーブル(不揮発性メモリ)に予め設定しておく必要があります。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘0’, ‘C’	位置決め動作起動指令
変更データ[H]	4	任意	ポジションスタート指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0 コントローラのポジション番号指定レジスタ(POSR:0D03_H)内のポジション番号の指定位置に移動を行う使用例を示します。

クエリ

1回目 01 05 04 0C FF 00 EB ...移動指令

2回目 01 05 04 0C 00 00 EA ...通常状態

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '0', 'C'	30343043
変更データ[H]	1回目: 'F', 'F', '0', '0' 2回目: '0', '0', '0', '0' (通常状態に戻してください。)	46463030 30303030
エラーチェック[H]	1回目: 'E', 'B' (LRC計算による) 2回目: 'E', 'A' (LRC計算による)	4542 4541
エンド	'CR', 'LF'	0D0A

ASCII
ファンクション
コード 05

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.9 ジョグ/インチング切替《JISL》

(1) 機能

ジョグとインチングの切替を行います。

ジョグ動作中に本ビットが切替わると減速停止します。

インチング動作中に本ビットが切替わってもインチング動作は継続されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘1’, ‘1’	ジョグ/インチング切替
変更データ[H]	4	任意	インチング動作状態: ‘F’, ‘F’, ‘0’, ‘0’ ジョグ動作状態: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをインテグレーション動作に切替ます。

クエリ:01 05 04 11 FF 00 E6

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '1'	30343131
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', '6' (LRC計算による)	4536
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.10 ティーチモード指令《MOD》

(1) 機能

通常運転モードと教示モードを切替えます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,D0への1点書き込み
開始アドレス[H]	4	'0', '4', '1', '4'	通常モード⇔教示モード切替
変更データ[H]	4	任意	教示モード:'F', 'F', '0', '0' 通常運転モード:'0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを教示モードに切替えます。

クエリ:01 05 04 14 FF 00 E3

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '4'	30343134
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', '3' (LRC計算による)	4533
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.11 ポジションデータ取込み指令《TEAC》

(1)機能

6.5.10 ティーチモード指令が FF00_H(教示モード)の時に、本指令(FF00_Hを書き込み)で現在位置データの取込みを行います。

取込み場所は、ポジション番号指定レジスタで指定されているポジション番号の中です。取込みポジションが空のポジションの場合、目標位置以外のデータ(位置決め幅 INP,速度 VCMD,加減速度 ACMD,制御フラグ CTLF)はパラメータの初期値と一緒に書込まれます。本指令(FF00_Hを書き込み)を行って20ms以上そのままの状態を保ってください。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,D0への1点書き込み
開始アドレス[H]	4	'0', '4', '1', '5'	ポジションデータ取込指令
変更データ[H]	4	任意	ポジションデータ取込指令 : 'F', 'F', '0', '0' 通常: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラが教示モード時に現在位置を取込みます。

クエリ:01 05 04 15 FF 00 E2

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '5'	30343135
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', '2' (LRC計算による)	4532
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.12 ジョグ+指令《JOG+》

(1) 機能

ジョグ またはインチング動作を行います。

・6.5.9 ジョグ／インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ+指令(変更データ FF00_H)を送信すると、反原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 のPIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ+指令(変更データ 0000_H)を送信するか、6.5.13 ジョグ-指令(変更データ FF00_H)を送信すると減速停止します。

・6.5.9 ジョグ／インチング切替指令が FF00_H(インチング設定)の時、ジョグ+指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、反原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘1’, ‘6’	ジョグ+指令
変更データ[H]	4	任意	ジョグ+指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ:01 05 04 16 FF 00 E1

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '6'	30343136
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', '1' (LRC計算による)	4531
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ

1回目 01 05 04 16 FF 00 E1 ...インチング移動

2回目 01 05 04 16 00 00 E0 ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '6'	30343046
変更データ[H]	1回目:'F', 'F', '0', '0' 2回目:'0', '0', '0', '0' (通常状態に戻してください。)	46463030 30303030
エラーチェック[H]	1回目:'E', '1' (LRC計算による) 2回目:'E', '0' (LRC計算による)	4531 4530
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.13 ジョグ-指令《JOG-》

(1) 機能

ジョグ またはインチング動作を行います。

・6.5.9 ジョグ／インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ-指令(変更データ FF00_H)を送信すると、原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 のPIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ-指令(変更データ 0000_H)を送信するか、6.5.12 ジョグ+指令(変更データ FF00_H)を送信すると減速停止します。

・6.5.9 ジョグ／インチング切替指令が FF00_H(インチング設定)の時、ジョグ-指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘1’, ‘7’	ジョグ指令
変更データ[H]	4	任意	ジョグ-指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ:01 05 04 17 FF 00 E0

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '7'	30343137
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'E', '0' (LRC計算による)	4530
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ

1回目 01 05 04 17 FF 00 E0 ...インチング移動

2回目 01 05 04 17 00 00 DF ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', '7'	30343047
変更データ[H]	1回目:'F', 'F', '0', '0' 2回目:'0', '0', '0', '0' (通常状態に戻してください。)	46463030 30303030
エラーチェック[H]	1回目:'E', '0' (LRC計算による) 2回目:'D', 'F' (LRC計算による)	4530 4446
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.14 スタートポジション0～7《ST0～ST7》(PIOパターン4、5 限定)

(1) 機能

指定されたポジションNo.位置に移動します。

スタートポジション0～7 移動指令は、PIOパターン4、5(電磁弁モード)が選択されている時に使用できます。

移動指令は、6.5.14(5)開始アドレス内の ST0～ST7 のどれかを有効にする(0000_Hの状態 で FF00_Hを書き込む)ことで行います。

有効スタートポジション以外を選択するとアラーム「085 移動時ポジションNo.異常」が発生します。ユーザーパラメータNo.27移動指令種別によりレベル動作とエッジ動作が選択可能です。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ～10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '5'	コイル,D0への1点書き込み
開始アドレス[H]	4	任意	6.5.14(5)開始アドレス参照
変更データ[H]	4	任意	※1動作指令: 'F', 'F', '0', '0' 動作指令: '0', '0', '0', '0'
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

※1 ユーザパラメータNo.27 移動指令種別を『レベル動作』設定した場合

FF00_H→0000_H書き込みで減速停止します。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをスタートポジション2へ移動します。

スタートポジション設定例

No	位置 [mm]	速度 [mm/s]	加速度 [G]	減速度 [G]
0	0.00	533.00	0.30	0.30
1	25.00	533.00	0.30	0.30
2	50.00	533.00	0.30	0.30

図6.2

クエリ

1回目 01 05 04 1D 00 00 D9...エッジを立てる為、0000_H書込み

2回目 01 05 04 1D FF 00 DA...移動指令

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '1', 'D'	30343044
変更データ[H]	1回目: '0', '0', '0', '0' 2回目: 'F', 'F', '0', '0'	30303030 46463030
エラーチェック[H]	1回目: 'D', '9' (LRC計算による) 2回目: 'D', 'A' (LRC計算による)	4439 4441
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

ASCII
ファンクション
コード 05

(5) 【開始アドレス】

アドレス	記号	名称	機能
0418	ST7	スタートポジション 7	ポジション 7 へ移動します
0419	ST6	スタートポジション 6	ポジション 6 へ移動します
041A	ST5	スタートポジション 5	ポジション 5 へ移動します
041B	ST4	スタートポジション 4	ポジション 4 へ移動します
041C	ST3	スタートポジション 3	ポジション 3 へ移動します
041D	ST2	スタートポジション 2	ポジション 2 へ移動します
041E	ST1	スタートポジション 1	ポジション 1 へ移動します
041F	ST0	スタートポジション 0	ポジション 0 へ移動します

6.5.15 ロードセルキャリブレーション指令《CLBR》専用ロードセル接続が必要

(1) 機能専用

専用ロードセルのキャリブレーションを行います。

ロードセルは工場出荷時、無負荷の状態を ON とするよう設定していますが、負荷を取付けた状態を基準(ON)としたい場合などには、キャリブレーションを行ってください。

その他にも必要な場合(再調整、点検等)、状況に応じて実施してください。

(2) ケリフォーマット

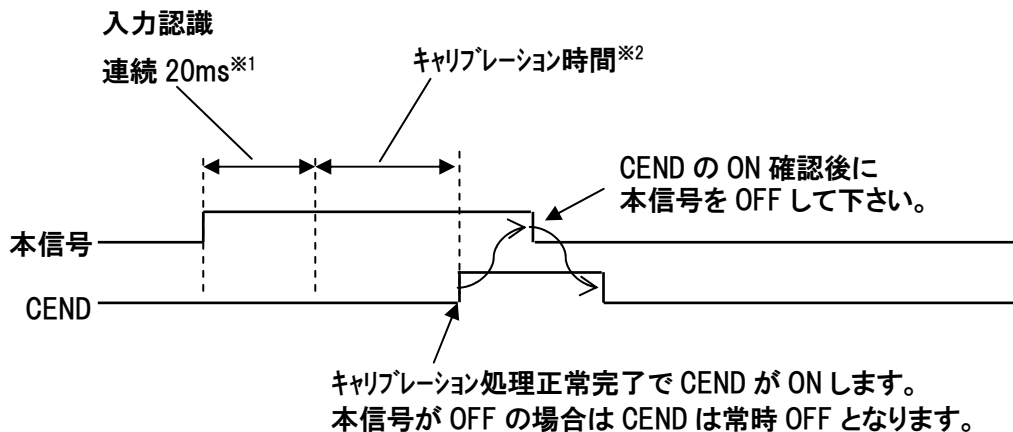
フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス[H]	2	‘0’, ‘4’, ‘2’, ‘6’	ロードセルキャリブレーション指令
変更データ[H]	2	任意	キャリブレーション実行指令:FF00 _H 通常運転時:0000 _H
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) キャリブレーション手順

- ① 運転を停止してください(軸動作中、押付け中、一時停止中はキャリブレーションできずにOE1:ロードセルキャリブレーション異常アラームとなります)。
- ② 本信号を 20ms 以上連続 ON してください。
- ③ キャリブレーションが完了するとキャリブレーション完了信号(4.3.2(12)デバイスステータスレジスタ1のCEND)が ON しますので、その後本信号を OFF してください。
キャリブレーションが正常に終了しなかった場合、OE1:ロードセルキャリブレーション異常アラームとなります。



注意:本信号が ON 状態では、通常運転指令は受け付けられません。



※1 この間に本信号を OFF した場合は、入力認識前のためキャリブレーション処理を行いません。

※2 この間に本信号を OFF した場合、アラームとなります。

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラに接続された専用ロードセルのキャリブレーションを行います。

クエリ 01 05 04 26 FF 00 D1

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '2', '6'	30343236
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'D', '1' (LRC計算による)	4431
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6. 5. 16 PIO/Modbus 切替設定 (PMSL)

(1) 機能

PIO外部指令信号の有効/無効の切替を行うことができます。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘2’, ‘7’	PIO/Modbus切替設定
変更データ[H]	4	任意	※1Modbus指令有効: ‘F’, ‘F’, ‘0’, ‘0’ Modbus指令無効: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※1 ・Modbus 指令有効(ON)(PIO 指令無効): FF00_H

PIO 信号による運転はできません。

・Modbus 指令無効(OFF)(PIO 指令有効): 0000_H

外部からの PIO 信号による運転が可能です。

補足 Modbus 指令を有効に変更した場合、変更時の PIO 状態が保持されています。
Modbus 指令を無効に切替えた場合、現在の PIO 状態により運転状態が変化します。但し、その時にエッジ検出で動作をする信号の状態が変化していてもエッジを検出したことにしないようにしています。

(3) 注意事項

- 運転モードスイッチ搭載機種では AUTO モードに変更されたら『PIO 指令有効』に
MANU モードに変更されたら『PIO 指令無効』になります。
- PIO 未搭載機種ではデフォルト設定が『PIO 指令無効』となります。
- 弊社ツール接続時(ティーチングペンダント、パソコン対応ソフト)は、ツール内モードには『ティーチモード 1, 2』、『モニターモード 1, 2』が存在します。この場合
『モニターモード 1, 2』 → 『PIO 指令有効』
『ティーチモード 1, 2』 → 『PIO 指令無効』 となります。

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるかもしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをModbus指令有効にします。

クエリ:01 05 04 27 FF 00 D0

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '2', '7'	30343237
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'D', '0' (LRC計算による)	4430
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6. 5. 17 減速停止《STOP》

(1) 機能

減速停止指令のエッジを立てる(FF00_Hを書込む)と、減速停止します。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス[H]	4	‘0’, ‘4’, ‘2’, ‘C’	減速停止設定
変更データ[H]	4	任意	減速停止指令(ON): ‘F’, ‘F’, ‘0’, ‘0’ ※コントローラが自動的に0000 _H にリセット します。
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、ケリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラに減速停止指令を出します。

クエリ:01 05 04 2C FF 00 CB

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '5'	3035
開始アドレス[H]	'0', '4', '2', 'C'	30343243
変更データ[H]	'F', 'F', '0', '0'	46463030
エラーチェック[H]	'C', 'B' (LRC計算による)	4342
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.6 制御情報の直接書き込み(使用ファンクションコード 06)

6.6.1 レジスタへの書き込み

※)6.2 [ASCIIコード表](#)
を参照ください。

(1) 機能

スレーブのレジスタの内容を変更(書き込み)します。

ブロードキャストの場合には、全スレーブの同じアドレスのレジスタの内容が変更されます。

[4.3.2 (5) デバイス制御レジスタ1 内容参照]

[4.3.2 (6) デバイス制御レジスタ2 内容参照]

[4.3.2 (7) ポジション番号指定レジスタおよびポジション移動指定レジスタ内容参照]

(2) 開始アドレス一覧

アドレス	記号	名称	バイト
0D00	DRG1	デバイス制御レジスタ1	2
0D01	DRG2	デバイス制御レジスタ2	2
0D03	POSR	ポジション番号指定レジスタ	2
9800	POSR	ポジション移動指定レジスタ	2

上記は制御指令のレジスタです。本レジスタのビットは、「PIO/Modbus 切替えステータス (PMSS)[4.3.2(14)参照]」が Modbus 指令無効(PIO 指令有効)の時、PIO パターンにより入力ポートに割り当てられます。本レジスタは Modbus 指令有効(PIO 指令無効)の時、書換えが可能です。

(3) クエリフォーマット

クエリメッセージでは、変更するレジスタのアドレスとデータを指定します。

変更したいデータは、クエリのデータエリアで16ビットのデータとして指定します。

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	'0', '6'	レジスタへの書込み
開始アドレス[H]	4	任意	6.6.1(2)開始アドレス一覧参照
変更データ[H]	4		4.3.2(5)~4.3.2(7) 変更データ 一覧参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

以下①～③に、それぞれの運転動作に応じた例を示します。

①軸No.0コントローラをサーボON→原点復帰させます。

クエリ

1回目 01 06 0D 00 10 00 DC...サーボON

2回目 01 06 0D 00 10 10 CC...原点復帰

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '6'	3036
開始アドレス[H]	'0', 'D', '0', '0'	30443030
変更データ[H]	1回目: '1', '0', '0', '0' 2回目: '1', '0', '1', '0'	31303030 31303130
エラーチェック[H]	1回目: 'D', 'C' (LRC計算による) 2回目: 'C', 'C' (LRC計算による)	4443 4343
エンド	'CR', 'LF'	0D0A

注1 サーボOFFの状態から 変更データを1010_Hとして送信しても原点復帰は行われません。

(各RCコントローラの起動時のタイミングチャート参照ください。)

注2 前の状態を維持したい場合は、変更がなくても前の状態を送信してください。

使用例のようにサーボONのビットは 原点復帰時も1のままにしておきます。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②ポジション移動指定レジスタ(アドレス9800_H)を使用してポジションNo.1に移動

前例①を行い、原点復帰を完了した状態で行ってください。

クエリ(前後にサイレントインターバルが入ります)

01 06 98 00 00 01 60

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '6'	3036
開始アドレス[H]	'9', '8', '0', '0'	39383030

変更データ[H]	'0', '0', '0', '1'	30303031
エラーチェック[H]	'6', '0' (LRC計算による)	3630
エンド	'CR', 'LF'	0D0A

注 本レジスタにポジションNo.を書き込むと移動を開始します。CSTR(スタート信号)は必要ありません。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

③ポジション番号指定レジスタ(アドレス0D03_H)を使用してポジションNo.1に移動

前例①を行い、原点復帰を完了した状態で行ってください。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 06 0D 03 00 01 E8 ...ポジションNo.1を指定

2回目 01 06 0D 00 10 00 DC ...CSTR(スタート信号)OFFを入力

3回目 01 06 0D 00 10 08 D4 ...CSTR(スタート信号)ONを入力

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ[H]
スタート	':'	3A
スレーブアドレス[H]	'0', '1'	3031
ファンクションコード[H]	'0', '6'	3036
開始アドレス[H]	1回目: '0', 'D', '0', '3' 2回目: '0', 'D', '0', '0' 3回目: '0', 'D', '0', '0'	30443033 30443030 30443030
変更データ[H]	1回目: '0', '0', '0', '1' 2回目: '1', '0', '0', '0' 3回目: '1', '0', '0', '8'	30303031 31303030 31303038
エラーチェック[H]	1回目: 'E', '8' (CRC計算による) 2回目: 'D', 'C' (CRC計算による) 3回目: 'D', '4' (CRC計算による)	4538 4443 4434
エンド	'CR', 'LF'	0D0A

注 前の状態を維持したい場合は、変更がなくても前の状態を送信してください。

使用例のようにサーボONのビットはサーボOFF以外では1のままにしてください。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.7 位置決めデータ直接書き込み(使用ファンクションコード 10)

6.7.1 直値移動指令

※) [6.2 ASCIIコード表](#)
を参照ください。

(1) 機能

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。アドレス 9900_H～9908_Hのレジスタ群を書換えるとアクチュエータに対し直値移動指令が行えます。(一伝文で送信が可能です。)

制御フラグ指定レジスタ(アドレス:9908_H)以外のレジスタは電源投入後、1度送信すればその後は有効となりますので、目標位置、位置決め幅、速度、加減速度、押し付け電流制限値、制御指定の変更の必要がない場合、その後は単独変更による実移動指令可能なレジスタ(開始アドレス一覧参照)の書込みだけで直値移動指令が可能となります。

(2) 開始アドレス一覧

目標位置座標、位置決め幅、速度、加減速度、押し付け電流制限値制御指定フラグ等を数値指定して移動を行うためのレジスタ群です。

開始アドレス一覧のデータ(合計6レジスタ)は、一度の送信で変更することが可能です。

アドレス [H]	記号	名称	符号	単独変更による 実移動指令 可能	レジスタ サイズ	バイト サイズ	単位
9900	PCMD	目標位置指定レジスタ	○	○	2	4	0.01mm
9902	INP	位置決め幅指定レジスタ		×	2	4	0.01mm
9904	VCMD	速度指定レジスタ		○	2	4	0.01mm/s
9906	ACMD	加減速度指定レジスタ		○	1	2	0.01G
9907	PPOW	押し付け時電流制限 指定レジスタ		○	1	2	%
9908	CTLF	制御フラグ指定レジスタ		×	1	2	—

ASCII
ファンクション
コード 10

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘1’, ‘0’	直値指令
開始アドレス[H]	4	任意	6.7.1(2)開始アドレス一覧参照
レジスタの数[H]	4	任意	6.7.1(2)開始アドレス一覧参照
バイト数[H]	2	上記レジスタ数による	上記指定レジスタ数の2倍の数値 を入力
変更データ1[H]	4		6.7.1(2)開始アドレス一覧参照
変更データ2[H]	4		6.7.1(2)開始アドレス一覧参照
変更データ3[H]	4		6.7.1(2)開始アドレス一覧参照
:	:		
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス[H]	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	2	‘1’, ‘0’	直値指令
開始アドレス[H]	4	任意	6.7.1(2)開始アドレス一覧参照
レジスタの数[H]	4	任意	6.7.1(2)開始アドレス一覧参照
エラーチェック[H]	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(5) レジスタ詳細説明

■ 目標位置指定レジスタ(PCMD)

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999(FFF0BDC1_H^(注1)～000F423F_H)です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると、ソフトリミット設定値の 0.2mm 手前^(注2)を目標位置として移動を開始します。目標位置座標指定レジスタ(記号: PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

注1 負の値は、2の補数で設定してください。

注2 インデックスモード設定の回転軸の場合はソフトリミット設定値が目標位置となります。

■ 位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって2種類の意味を持ちます。1つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、1～999999(1_H～000F423F_H)です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

本レジスタを変更しただけでは移動開始は行われません。

■ 速度指定レジスタ(VCMD)

移動速度を指定します。単位は 0.01mm/s で設定範囲は 1～999999(1_H～000F423F_H)です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタの下位ワードが書換えられると移動開始します。つまり移動中の速度可変を行う場合、本レジスタを書換えることにより実現できます。

■加減速度指定レジスタ(ACMD)

加速度・減速度を指定します。単位は 0.01G で設定範囲は 1~300(1_H~012C_H) です。ただし、パラメータの最大加速度および最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタが書換えられると移動開始します。つまり移動中の加減速度可変を行う場合、本レジスタを書換えることにより実現できます。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。以下を参照して設定してください。

アクチュエータ型式	押付け可能範囲[%]	設定可能範囲(入力値)[H]
RCS2-RA13R 以外の アクチュエータ	20~70 ^(注)	33~B2
RCS2-RA13R	20~200	33~1FE

注 各アクチュエータによっては設定範囲が異なる場合があります。

[詳しくは当社カタログ、または取扱説明書を参照]

本レジスタが書換えられると移動開始します。つまり押付け動作中の電流制限値可変を行う場合、本レジスタを書換えることにより実現できます。

押付け電流値設定例

●20%設定例

$$255(100\%) \times 0.2(20\%) = 51 \rightarrow 33_{\text{H}}(16 \text{ 進数変換})$$

■制御フラグ指定レジスタ(CTLF)

動作方法を設定します。

押付け動作およびインクリメンタル動作(ピッチ送り)の場合、本レジスタを移動指令のたびに設定してください(1回の移動のたびに初期値で書き換えられます)。

CTLF ビット構成

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
MSB	-	-	NTC1	NTC0	-	-	-	-	MOD1	MOD0	GSL1	GSL0	INC	DIR	PUSH	-	LSB

ビット 1(PUSH) = 0:通常動作(初期値)

1:押付け動作

ビット 2(DIR) = 0:アプローチ動作完了後の押付け動作の方向を正転とします(初期値)。

1:アプローチ動作完了後の押付け動作の方向を逆転とします。

このビットにより、PCMD からの最終停止位置の方向を算出しますので方向を間違えると下図 6.3 のように(2×INP)の幅だけずれた動作になりますので、注意してください。

また、ビット 1 の設定値が 0 の場合はこのビットの設定値は無効です。

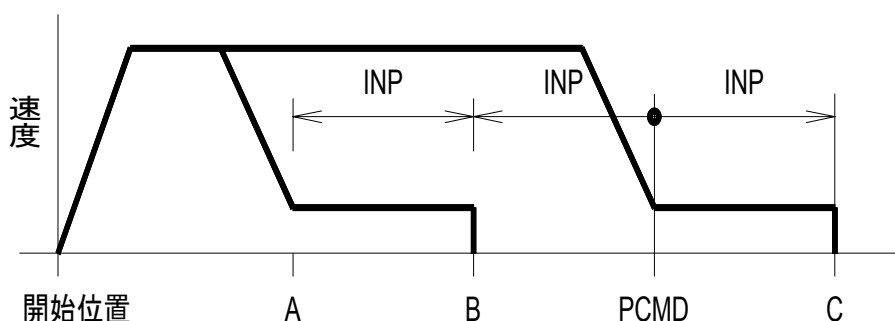


図 6.3 押付け時の動作方向

ビット 3(INC) = 0:通常動作(初期値)

1:インクリメンタル動作(ピッチ送り)

このビットを 1 に設定することにより、相対位置移動を行う事が可能となります。この動作は、通常動作時と押付け動作時(CTLF のビット 1)で挙動が異なります。通常動作時は目標位置(PCMD)に対しての移動量を生成するのに対し、押付け動作時(ビット 1=1 の時)には現在位置に対しての移動量を生成します。

なお、相対座標の計算は mm 単位加算後パルス変換を行うので、パルス変換

後の加算方法の場合に発生する『相対移動を繰り返した場合は、リート設定による割り切れないパルスが累積誤差として位置ずれを起こす現象』は発生しません。

ビット 4(GSL0)、5(GSL1)= 以下の表参照(SCON-CA/CFA だけ設定可能)

GSL1	GSL0	機能
0	0	パラメータセット 0 選択(初期値)
0	1	パラメータセット 1 選択
1	0	パラメータセット 2 選択
1	1	パラメータセット 3 選択

あらかじめ 6 種のサーボゲインパラメータを最大 4 セット登録しておくことで、ポジション移動ごとに選択したパラメータセットで運転を行います。[詳細はコントローラ本体の取扱説明書参照]

ビット 6(MOD0)、7(MOD1)= 以下の表参照 (PCON-*, ERC2/ERC3 は、設定できません)

MOD1	MOD0	機能
0	0	台形パターン(初期値)
0	1	S 字モーション
1	0	一次遅れフィルタ
1	1	使用できません

加減速パターン特性を選択するための信号です。いずれかをアクチュエータの移動指令前に選択してください。[詳細はコントローラ本体の取扱説明書参照]

ビット 12(NTC0)、13(NTC1)= 以下の表参照(SCON-CA だけ設定可能)

NTC1	NTC0	機能
0	0	制振制御を使用しない(初期値)
0	1	パラメータセット 1 選択
1	0	パラメータセット 2 選択
1	1	パラメータセット 3 選択

制振制御の使用選択 および、あらかじめパラメータを最大 3 セット登録しておくことで、ポジション移動ごとに選択したパラメータセットで運転を行います。[詳細はコントローラ本体の取扱説明書参照]

(6) 使用例

以下①～⑦に、それぞれの運転動作に応じた例を示します。

① 目標位置を変更して移動する(目標位置以外のデータは、パラメータ初期値を使用)

条件:コントローラユーザーパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値
で動作条件は OK。目標位置だけを変更しアクチュエータを動作させたい。

補足:コントローラユーザーパラメータについて

- ・速度初期値(パラメータNo.8)→カタログ内の該当アクチュエータ最高速度
- ・加減速度初期値(パラメータNo.9)→カタログ内の該当アクチュエータ定格加速度
- ・位置決め幅初期値(パラメータNo.10)→デフォルト値 0.1mm

目標位置指定レジスタ(9900_H)の書込み^(例1)



(例1) 目標位置を 50mm の位置とする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	設定不要				

■クエリ :01 10 9900 0002 04 0000 1388 B5[CR][LF]

■レスポンス :01 10 9900 0002 54[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	':'	3A	
スレーブアドレス	'0', '1'	3031	軸No.0+1
ファンクションコード	'1', '0'	3130	
開始アドレス	'9', '9', '0', '0'	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	'0', '0', '0', '2'	30303032	アドレス9900 _H ～9901 _H まで 書込み指定
バイト数	'0', '4'	3034	2(レジスタ)×2=4(バイト)→4 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'1', '3', '8', '8'	31333838	50[mm]×100=5000→1388 _H
エラーチェック	'B', '5'	4235	LRCチェック計算結果→B5 _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	27		

ASCII
ファンクション
コード 10

②目標位置を変更して移動する(目標位置以外のデータも変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させたい

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み(例2)



移動開始

(例2) 目標位置を 50mm の位置とする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	0.1	100	0.3		設定不要

■クエリ :01 10 9900 0007 0E 0000 1388 0000 000A 0000 2710 001E 47[CR][LF]

■レスポンス :01 10 9900 0007 4F[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	':'	3A	
スレーブアドレス	'0', '1'	3031	軸No.0+1
ファンクションコード	'1', '0'	3130	
開始アドレス	'9', '9', '0', '0'	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	'0', '0', '0', '7'	30303039	アドレス9900 _H ～9906 _H まで 書込み指定
バイト数	'0', 'E'	3132	7(レジスタ)×2=14(バイト)→E _H
変更データ1、2(目標位置) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'1', '3', '8', '8'	31333838	50[mm]×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'0', '0', '0', 'A'	30303041	0.1[mm]×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'2', '7', '1', '0'	32373130	100[mm/s]×100=10000 →2710 _H
変更データ7(加減速度) 入力単位(0.01G)	'0', '0', '1', 'E'	30303145	0.3[G]×100=30→001E _H
エラーチェック	'4', '7'	3437	LRCチェック計算結果→47 _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	47		

③移動中の速度変更を行う

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み^(例 2)



移動開始



速度指定レジスタ(9904_H～9905_H)の書込み^(例 3)



変更された速度で引き続き移動

(例3) 移動中に速度 100mm/s→50mm/s に変更する場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	0.1	100→50	0.3		設定不要

(1)速度 100mm/s で移動開始[前例の②目標位置を変更して移動するを参照]

■クエリ :01 10 9900 0007 0E 0000 1388 0000 000A 0000 2710 001E 47[CR][LF]

■レスポンス :01 10 9900 0007 4F[CR][LF]

(2)速度 50mm/s に速度変更

■クエリ :01 10 9904 0002 04 0000 1388 B1[CR][LF]

■レスポンス :01 10 9904 0002 50[CR][LF]

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳(50mm/s に速度変更 [100mm/s で移動開始は前例参照])

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘9’, ‘9’, ‘0’, ‘4’	39393034	開始アドレスは目標位置指定 レジスタ9904 _H から
レジスタの数	‘0’, ‘0’, ‘0’, ‘2’	30303032	アドレス9904 _H ～9905 _H まで 書込み指定
バイト数	‘0’, ‘4’	3034	2(レジスタ)×2=4(バイト)→4 _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘1’, ‘3’, ‘8’, ‘8’	31333838	50[mm/s]×100=5000 →1388 _H
エラーチェック	‘B’, ‘1’	4231	LRCチェック計算結果→B1 _H
トレーラ	‘CR’, ‘LF’	0D0A	
合計バイト数	27		

④インクリメンタル移動(ピッチ送り)の場合

条件:コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値
/で動作条件は OK。ピッチ幅だけを変更しアクチュエータを動作させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例 4)



移動開始

補足:アドレス 9900_Hと 9908_Hだけを一度のデータ送信で変更することはできません。アドレスは連番となっているため、9900_Hと 9908_Hだけを変更したい場合は、2 度の伝文送信で実施してください。

1 度の伝文送信で済ませたい場合は、9900_H～9908_H までを全て書込み実行を行ってください。

(例4) ピッチ幅を 10mm に設定してインクリメンタル移動する場合

ピッチ幅 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
10	0.1	100	0.3	0	インクリメンタル (bit3=1)

■クエリ :01 10 9900 0009 12 0000 03E8 0000 000A 0000 2710 001E 0000 0008 E9[CR][LF]

■レスポンス :01 10 9900 0009 4D[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘9’, ‘9’, ‘0’, ‘0’	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	‘0’, ‘0’, ‘0’, ‘9’	30303039	アドレス9900 _H ~9908 _H まで 書き込み指定
バイト数	‘1’, ‘2’	3132	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘0’, ‘3’, ‘E’, ‘8’	30334538	10[mm]×100=1000→03E8 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘0’, ‘0’, ‘0’, ‘A’	30303041	0.1[mm]×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘2’, ‘7’, ‘1’, ‘0’	32373130	100[mm/s]×100=10000 →2710 _H
変更データ7(加減速度) 入力単位(0.01G)	‘0’, ‘0’, ‘1’, ‘E’	30303145	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	‘0’, ‘0’, ‘0’, ‘0’	30303030	0[%]→0 _H
変更データ9(制御フラグ)	‘0’, ‘0’, ‘0’, ‘8’	30303038	(インクリメンタル設定) 1000b→0008 _H
エラーチェック	‘E’, ‘9’	4539	LRCチェック計算結果→E9 _H
トレーラ	‘CR’, ‘LF’	0D0A	
合計バイト数	55		

⑤インクリメンタル移動(ピッチ送り)中に速度変更を行う場合

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを相対位置決め動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例4)



インクリメンタル移動開始



速度指定レジスタ(9904_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み^(例5)



変更された速度で引き続きインクリメンタル移動

補足: 制御フラグ指定レジスタ(9908_H)は設定後、一度アクチュエータが動作開始すると初期値(0_H:通常移動)に戻ります。このため、インクリメンタルおよび押付け動作を再度行う場合は、制御フラグ指定レジスタ(9908_H)を再設定し、送信する必要があります。

(例 5) 移動中に速度 100mm/s→50mm/s に変更する場合

ピッチ幅 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
10	0.1	100→50	0.3	0	インクリメンタル (bit3=1)

(1)速度 100mm/s で移動開始[前例の④インクリメンタル移動(ピッチ送り)の場合を参照]

■クエリ :01 10 9900 0009 12 0000 03E8 0000 000A 0000 2710 001E 0000 0008 E9[CR][LF]

■レスポンス :01 10 9900 0009 4D[CR][LF]

(2)速度 50mm/s に速度変更

■クエリ :01 10 9904 0005 0A 0000 1388 001E 0000 0008 82[CR][LF]

■レスポンス :01 10 9904 0005 4D[CR][LF]

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳(50mm/s に速度変更 [100mm/s で移動開始は前例参照])

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	':'	3A	
スレーブアドレス	'0', '1'	3031	軸No.0+1
ファンクションコード	'1', '0'	3130	
開始アドレス	'9', '9', '0', '4'	39393034	開始アドレスは目標位置指定 レジスタ9904 _H から
レジスタの数	'0', '0', '0', '5'	30303032	アドレス9904 _H ~9908 _H まで 書込み指定
バイト数	'0', 'A'	3034	5(レジスタ)×2=10(バイト)→A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'1', '3', '8', '8'	31333838	50[mm/s]×100=5000 →1388 _H
変更データ7(加減速度) 入力単位(0.01G)	'0', '0', '1', 'E'	30303145	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	'0', '0', '0', '0'	30303030	0[%]→0 _H
変更データ9(制御フラグ)	'0', '0', '0', '8'	30303038	(インクリメンタル設定) 1000b→0008 _H
エラーチェック	'8', '2'	3832	LRCチェック計算結果→82 _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	39		

⑥押付け動作の場合(押付け動作中の押付け力の変更)

条件: 押付け動作をさせたい。但し、押し付け中には任意のタイミングで押付け力を変化させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:
押付け設定)の書込み^(例6)



押付け動作開始



押付け動作中に押付け電流制限指定レジスタ(9907_H)～制御
フラグ指定レジスタ(9908_H:押付け設定)の書込み^(例7)



変更された押付け力で引き続き押付け動作

(例6) 50mmの位置から20mmの幅で押付け電流制限70%の押付け動作をする場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	20	100	0.3	70	押付け動作 (bit1=1、 Bit2=0、1)

■クエリ :01 10 9900 0009 12 0000 1388 0000 07D0 0000 2710 001E 00B2 0006 BC[CR][LF]

■レスポンス :01 10 9900 0009 4D[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘9’, ‘9’, ‘0’, ‘0’	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	‘0’, ‘0’, ‘0’, ‘9’	30303039	アドレス9900 _H ~9908 _H まで 書込み指定
バイト数	‘1’, ‘2’	3132	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2(目標位置) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘1’, ‘3’, ‘8’, ‘8’	31333838	50[mm]×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘0’, ‘7’, ‘D’, ‘0’	30374430	20[mm]×100=2000→07D0 _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘2’, ‘7’, ‘1’, ‘0’	32373130	100[mm/s]×100=10000 →2710 _H
変更データ7(加減速度) 入力単位(0.01G)	‘0’, ‘0’, ‘1’, ‘E’	30303145	0.3[G]×100=30→001E _H
変更データ8(押付け) 入力単位[%]	‘0’, ‘0’, ‘B’, ‘2’	30304232	70[%]→B2 _H
変更データ9(制御フラグ)	‘0’, ‘0’, ‘0’, ‘6’	30303036	(押付け設定:) 0110b→0006 _H
エラーチェック	‘B’, ‘C’	4243	LRCチェック計算結果→BC _H
トレーラ	‘CR’, ‘LF’	0D0A	
合計バイト数	55		

(例7) 押付け動作中に押付け電流制限を70%→50%に変更する場合

目標位置 [mm]	位置決め幅 [mm]	速度 [mm/s]	加減速度 [G]	押付け [%]	制御フラグ
50	20	100	0.3	70→50	押付け動作 (bit1=1、 bit2=1)

■クエリ :01 10 9907 0002 04 007F 0006 C4[CR][LF]

■レスポンス :01 10 9907 0002 4D[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘9’, ‘9’, ‘0’, ‘7’	39393037	開始アドレスは目標位置指定 レジスタ9907 _H から
レジスタの数	‘0’, ‘0’, ‘0’, ‘2’	30303032	アドレス9907 _H ～9908 _H まで 書込み指定
バイト数	‘0’, ‘4’	3034	2(レジスタ)×2=4(バイト)→4 _H
変更データ8(押付け) 入力単位[%]	‘0’, ‘0’, ‘7’, ‘F’	30303746	50[%]→7F _H
変更データ9(制御フラグ)	‘0’, ‘0’, ‘0’, ‘6’	30303036	(押付け設定:) 0110b→0006 _H
エラーチェック	‘C’, ‘4’	4334	LRCチェック計算結果→C4 _H
トレーラ	‘CR’, ‘LF’	0D0A	
合計バイト数	27		

ASCII
ファンクション
コード 10

⑦注意事項(移動中の位置決め幅の変更)

移動中の位置決め幅の変更はできません。

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングで位置決め幅を変更をしたい
(変更できません。書込みを行った場合、このデータは次の位置決めに反映されることとなります。)

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み



通常動作開始



位置決め幅指定レジスタ(9902_H～9903_H)の書込み



変更前の位置決め幅設定で引き続き通常動作

補足: 位置決め幅指定レジスタは単独書込みによる実移動指令は無効です。
このため位置決め幅指定レジスタ(9902_H～9903_H)の書込みによるデータは次回移動指令を実施した際に有効となります。

6.7.2 ポジションテーブルデータ書込み

(1) 機能

このクエリを使用することによってポジションテーブル上のデータ変更が可能です。

開始アドレス一覧(アドレス+0000_H～+000E_H)にアクセスがある毎に1ポジションデータ単位で吹き不揮発性メモリ(EEPROM、FeRAM)から読出され、書込み実施後、不揮発性メモリに再び格納されます。各コントローラ取扱説明書の基本仕様から書込み回数制限を確認してください。

※EEPROMはデバイスの制約上、書込み回数が約10万回と制限されています。ポジションテーブルデータの書換えを頻繁に行くと短時間でEEPROMの書換え回数をオーバーし故障の原因となりますので、上位側のロジックは想定外のループ等が発生しないようご注意ください。

FeRAMは書込み回数に制限がありません。

(2) 開始アドレス一覧

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_{\text{H}} + (16 \times \text{ポジション No.})_{\text{H}} + \text{アドレス(オフセット値)}_{\text{H}}$$

(例) ポジションNo.200 の速度指令レジスタを変更したい場合

$$\begin{aligned} & 1000_{\text{H}} + (16 \times 200 = 3200)_{\text{H}} + 4_{\text{H}} \\ & = 1000_{\text{H}} + \text{C80}_{\text{H}} + 4_{\text{H}} \\ & = 1\text{C84}_{\text{H}} \end{aligned}$$

『1C84』がクエリ開始アドレス部入力値になります。

注 最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

■ポジションデータ変更レジスタ群

アドレス	記号	名称	符号	レジスタ サイズ	バイト サイズ	入力 単位
+0000	PCMD	目標位置	○	2	4	0.01mm
+0002	INP	位置決め幅		2	4	0.01mm
+0004	VCMD	速度指令		2	4	0.01mm/s
+0006	ZNMP	個別ゾーン境界+側	○	2	4	0.01mm
+0008	ZNLP	個別ゾーン境界-側	○	2	4	0.01mm
+000A	ACMD	加速度指令		1	2	0.01G
+000B	DCMD	減速度指令		1	2	0.01G
+000C	PPOW	押付け時電流制限値		1	2	%
+000D	LPOW	負荷電流閾値		1	2	%
+000E	CTLF	制御フラグ指定		1	2	

※ '+' が付いているアドレスはオフセット値です。

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	ASCIIモード 固定文字列	文字数 (バイト数)	備考
ヘッダ	‘:’	1	
スレーブアドレス[H]	任意	2	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	‘1’, ‘0’	2	
開始アドレス[H]	任意	4	6.7.2(2)開始アドレス参照一覧
レジスタの数[H]	任意	4	6.7.2(2)開始アドレス参照一覧
バイト数[H]	上記レジスタ数に よる	2	上記指定レジスタ数の2倍の数値 を入力
変更データ1[H]		4	6.7.2(2)開始アドレス参照一覧
変更データ2[H]		4	6.7.2(2)開始アドレス参照一覧
変更データ3[H]		4	6.7.2(2)開始アドレス参照一覧
:		:	
エラーチェック[H]	LRC計算結果	2	
トレーラ	‘CR’, ‘LF’	2	
合計バイト数		最大256	

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	ASCIIモード 固定文字列	文字数 (バイト数)	備考
ヘッダ	‘:’	1	
スレーブアドレス[H]	任意	2	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード[H]	‘1’, ‘0’	2	
開始アドレス[H]	任意	4	6.7.2(2)開始アドレス参照一覧
レジスタの数[H]	任意	4	6.7.2(2)開始アドレス参照一覧
エラーチェック[H]	LRC計算結果	2	
トレーラ	‘CR’, ‘LF’	2	
合計バイト数		17	

(5) レジスタ詳細説明

■目標位置(PCMD)

ポジション移動時の位置決め目標位置を絶対座標上の位置、または相対距離で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999(FFF0BDC1_H^(注)～000F423F_H)です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると、ソフトリミット設定値の 0.2mm 手前^(注2)を目標位置として移動を開始します。目標位置座標指定レジスタ(記号:PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

(注1)負の値は、2 の補数で設定してください。

(注2)インデックスモード設定の回転軸の場合はソフトリミット設定値が目標位置となります。

■位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって 2 種類の意味を持ちます。1 つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2 つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、1～999999(1_H～000F423F_H)です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■速度指定レジスタ(VCMD)

ポジション移動時の、移動速度を指定します。単位は 0.01mm/s で設定範囲は 1～999999 (1_H～000F423F_H)です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■個別ゾーン境界±(ZNMP,ZNLP)

パラメータで設定されるゾーン境界とは別に、ポジション移動時のみ有効なゾーン信号を出力します。

絶対位置座標で表現した+側のゾーン信号出力境界値を ZNMP に、-側のゾーン信号出力境界値を ZNLP に設定します。現在位置がこの±境界値の内側にあるときは、ゾーンステータスレジスタの対応するビットが ON となります。設定単位は 0.01mm となります。設定可能範囲はともに -999999～999999(FFF0BDC1_H^(注)～000F423F_H) となります。

個別ゾーン出力を無効にする場合は、ZNMP と ZNLP を同じ値としてください。

注 負の値は、2 の補数で設定してください。

■ 加速度指定レジスタ(ACMD)

ポジション移動時の、加速度を指定します。単位は 0.01G で設定範囲は 1~300 (1_H~012C_H) です。ただし、パラメータの最大加速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■ 減速度指定レジスタ(DCMD)

ポジション移動時の、減速度を指定します。単位は 0.01G で設定範囲は 1~300 (1_H~012C_H) です。ただし、パラメータの最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■ 押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。以下を参照して設定してください。

アクチュエータ型式	押付け可能範囲[%]	設定可能範囲(入力値)[H]
RCS2-RA13R 以外の アクチュエータ	20~70 ^(注)	33~B2
RCS2-RA13R	20~200	33~1FE

注 各アクチュエータによっては設定範囲が異なる場合があります。

[詳しくは当社カタログ、または取扱説明書を参照]

押付け電流値設定例

●20%設定例

$255(100\%) \times 0.2(20\%) = 51 \rightarrow 33_{\text{H}}$ (16 進数変換)

■ 負荷出力電流閾値(LPOW)

負荷出力判定を行う場合、電流閾値を LPOW に設定します。設定は、押付け時電流制限値(PPOW)同様に使用するアクチュエータに合わせて行います。判定を行わない場合は 0 を設定してください。

■ 制御フラグ指定レジスタ(CTLF)

[6.7.1(5) 制御フラグ指定レジスタ参照]

(6) 使用例

軸No.0 のポジションNo.12 全データを以下の様書き換えます。

目標位置 [mm]	位置 決め幅 [mm]	速度 [mm/s]	個別ゾーン 境界+側 [mm]	個別ゾーン 境界-側 [mm]	加速度 [G]	減速度 [G]	押付け [%]	閾値	制御フラグ
100	0.1	200	60	40	0.01	0.3	0	0	通常移動

■ケリ:01 10 10C0 000F 1E 0000 2710 0000 000A 0000 4E20 0000 1770
0000 0FA0 0001 001E 0000 0000 0000 EE[CR][LF]

■受信レスポンス :01 10 10C0 000F 10[CR][LF]

■ケリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
ヘッダ	':'	3A	
スレーブアドレス	'0', '1'	3031	軸No.0+1
ファンクションコード	'1', '0'	3130	
開始アドレス	'1', '0', 'C', '0'	31304330	開始アドレスはポジションNo.12における目標位置指定レジスタ10C0 _H から※1
レジスタの数	'0', '0', '0', 'F'	30303046	レジスタ記号PCMD~CTLFまで 合計15レジスタ書き込み指定
バイト数	'1', 'E'	3145	15(レジスタ)×2=30(バイト)→1E _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	'0', '0', '0', '0' '2', '7', '1', '0'	30303030 32373130	32ビットデータ上位ビットは全て0 100[mm]×100=10000→2710 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	'0', '0', '0', '0' '0', '0', '0', 'A'	30303030 30303041	32ビットデータ上位ビットは全て0 0.1[mm]×100=10→000A _H
変更データ5、6 (速度) 入力単位(0.01mm/SEC)	'0', '0', '0', '0' '4', 'E', '2', '0'	30303030 34453230	32ビットデータ上位ビットは全て0 200[mm/s]×100=20000 →4E20 _H
変更データ7、8 (個別ゾーン境界+) 入力単位(0.01mm)	'0', '0', '0', '0' '1', '7', '7', '0' '0', 'F', 'A', '0'	30303030 31373730 30464130	32ビットデータ上位ビットは全て0 60[mm]×100=6000→1770 _H 40[mm]×100=4000→0FA0 _H

ASCII
ファンクション
コード 10

次ページに続く

前ページからの続き

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ[H]	備考
変更データ9、10 (個別ゾーン境界) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'0', 'F', 'A', '0'	30464130	40[mm]×100=4000→0FA0 _H
変更データ11(加速度) 入力単位(0.01G)	'0', '0', '0', '1'	30303031	0.01[G]×100=1→0001 _H
変更データ12(減速度) 入力単位(0.01G)	'0', '0', '1', 'E'	30303145	0.3[G]×100=30→001E _H
変更データ13(押付け) 入力単位[%]	'0', '0', '0', '0'	30303030	0[%]→0 _H
変更データ14(閾値) 入力単位[%]	'0', '0', '0', '0'	30303030	0[%]→0 _H
変更データ15(制御フラグ)	'0', '0', '0', '0'	30303030	通常動作のため全て 0 0000b→0000 _H
エラーチェック	'E', 'E'	4545	LRCチェック計算結果→EE _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	79		

※1) 開始アドレス計算

例 ポジションNo.12 の全データを変更をかける為、本クエリの開始アドレス部には
ポジションNo.12 の目標位置アドレスとなる。

$$1000_{\text{H}} + (16 \times 12 = 192)_{\text{H}} + 0_{\text{H}}$$

$$= 1000_{\text{H}} + \text{C0}_{\text{H}} + 0_{\text{H}}$$

$$= 10\text{C0}_{\text{H}}$$

『10C0』が本クエリ開始アドレス部入力値になります。

以下は当社 RC 用パソコン対応ソフト上ポジションデータ内で、クエリメッセージ送信前と後の違いを表示したものです。

(注)パソコンソフトと Modbus の両方共を同時に接続することはできません。次の例は、パソコンソフトと Modbus の接続を切り替えて行った例です。

■クエリ送信前

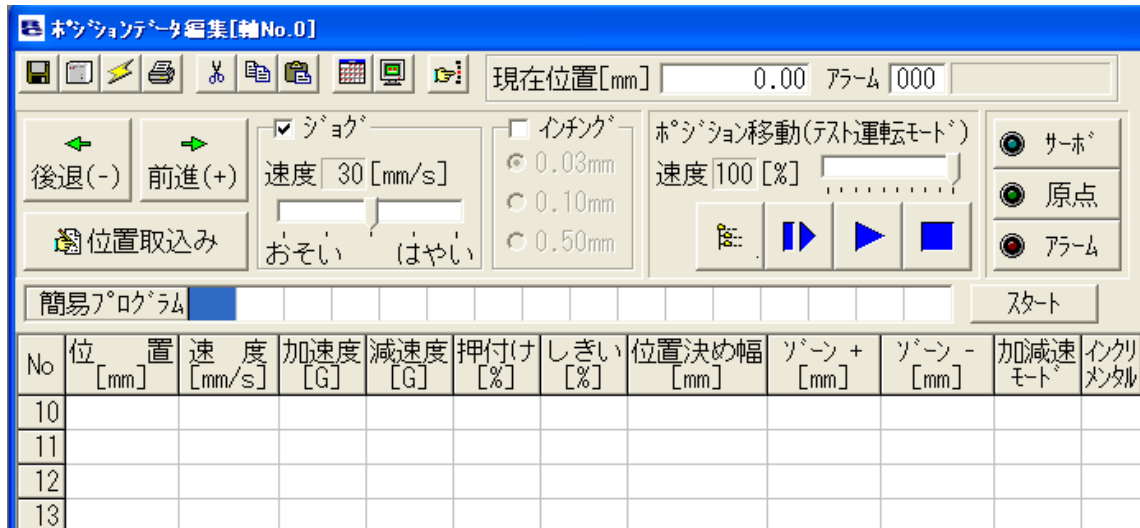



図 6.4

■送信後



図 6.5

※  ボタンを押すか、ポジションデータ編集画面を開き直さないと書き換えた内容は表示されません。

ASCII
ファンクション
コード 10

7 トラブルシューティング



7. 1 異常時の返信(例外レスポンス)について

ブロードキャスト・メッセージ以外のクエリ(命令)の場合、マスターは正常なレスポンス(返信)を期待してクエリを発行します。それに対して、スレーブはレスポンスを返さなければなりません。正常な場合には正常レスポンスを返しますが、異常が発生した場合、例外レスポンスを返します。

クエリに対するスレーブの応答の方法には、以下の4種類があります。

- (1) 正常にクエリを受信し、正常に処理を行い、正常レスポンスを返します。
- (2) 通信エラー等により、スレーブはクエリを受信することができなく、無応答になります。マスターはタイムアウトによるエラー検出を行います。
- (3) スレーブはクエリを受信することができたが、LRC/CRC エラーを検出し、正しいクエリではない場合には、スレーブは無応答になります。従って、マスターはタイムアウトによるエラー検出を行います。
- (4) スレーブは、エラーもなく正しいクエリを受信できたが、何らかの理由(例えば、当該レジスタは存在しないなど)で処理できない場合には、例外の内容を示す例外コードをつけて例外レスポンスを返します。

例外レスポンスが発生する場合の例

(Read Input Status のクエリメッセージ例)

フィールド名称	数値例 (Hex)	ASCIIモード 文字列	RTUモード 8ビット(Hex)
ヘッダ		':'	なし
スレーブアドレス	03 _H	'0', '3'	03 _H
ファンクションコード	02 _H	'0', '2'	02 _H
開始アドレス(上位)	04 _H	'0', '4'	04 _H
開始アドレス(下位)	A1 _H	'A', '1'	A1 _H
DIの数(上位)	00 _H	'0', '0'	00 _H
DIの数(下位)	14 _H	'1', '4'	14 _H
エラーチェック		LRC(2文字)	CRC(16ビット)
トレーラ		CR / LF	なし
	合計バイト数	17	8

入力ステータス04A1_Hが存在しない場合、以下の例外レスポンスが返されます。

スレーブからの例外レスポンス例

フィールド名称	数値例 (Hex)	ASCIIモード 文字列	RTUモード 8ビット(Hex)
ヘッダ		':'	なし
スレーブアドレス	03 _H	'0', '3'	03 _H
ファンクションコード	82 _H	'8', '2'	82 _H
例外コード	02 _H	'0', '2'	02 _H
エラーチェック		LRC(2文字)	CRC(16ビット)
トレーラ		CR / LF	なし
	合計バイト数	11	5

例外レスポンスは、スレーブアドレス、ファンクションコードおよびデータフィールドより構成されています。スレーブアドレスフィールドには、正常レスポンスと同じように、スレーブのアドレスをセットします。ファンクションコードフィールドには、クエリのファンクションコードをセットし更に、そのMSB(ファンクションコードの最上位ビット)を1にします。これによりマスタは、正常レスポンスではなく例外レスポンスであることを検知できます。データフィールドには、例外の内容を示す例外コードがセットされます。

例) クエリファンクションコード 『02_H』(00000010_b)

→例外レスポンスファンクションコード 『82_H』 (10000010_b)

■例外コード

RC シリーズコントローラで発生する、例外コード及び内容を示します。

コード (Hex)	例外コード名	機能	備考
01 _H	Illegal Function	不正ファンクション	ファンクションの誤りによりスレーブ側が重度のエラー発生し、クエリの実行ができないとき
02 _H	Illegal Data Address	不正データアドレス	データアドレスが許されない値の時
03 _H	Illegal Data Value	不正データ	データ値が許されない値の時
04 _H	Slave Device Failure	スレーブ回復不能エラー発生 の為、実行不可	スレーブ側の重度エラー発生により、クエリの実行ができないとき

7.2 注意事項

- ・ Modbus ファンクションによってレジスタを参照する場合、1 伝文で複数のカテゴリのレジスタを同時に読出すことは出来ません。従って、カテゴリをまたいで参照する場合は、カテゴリの区切りのアドレスで複数伝文に分けて読み出すようにしてください。
- ・ 本仕様書は RC コントローラシリーズ「プロトコル M」搭載機種共通の説明となっています。機種固有の仕様等は各 RC コントローラの取扱説明書を参照ください。

7.3 通信がうまくいかない時は

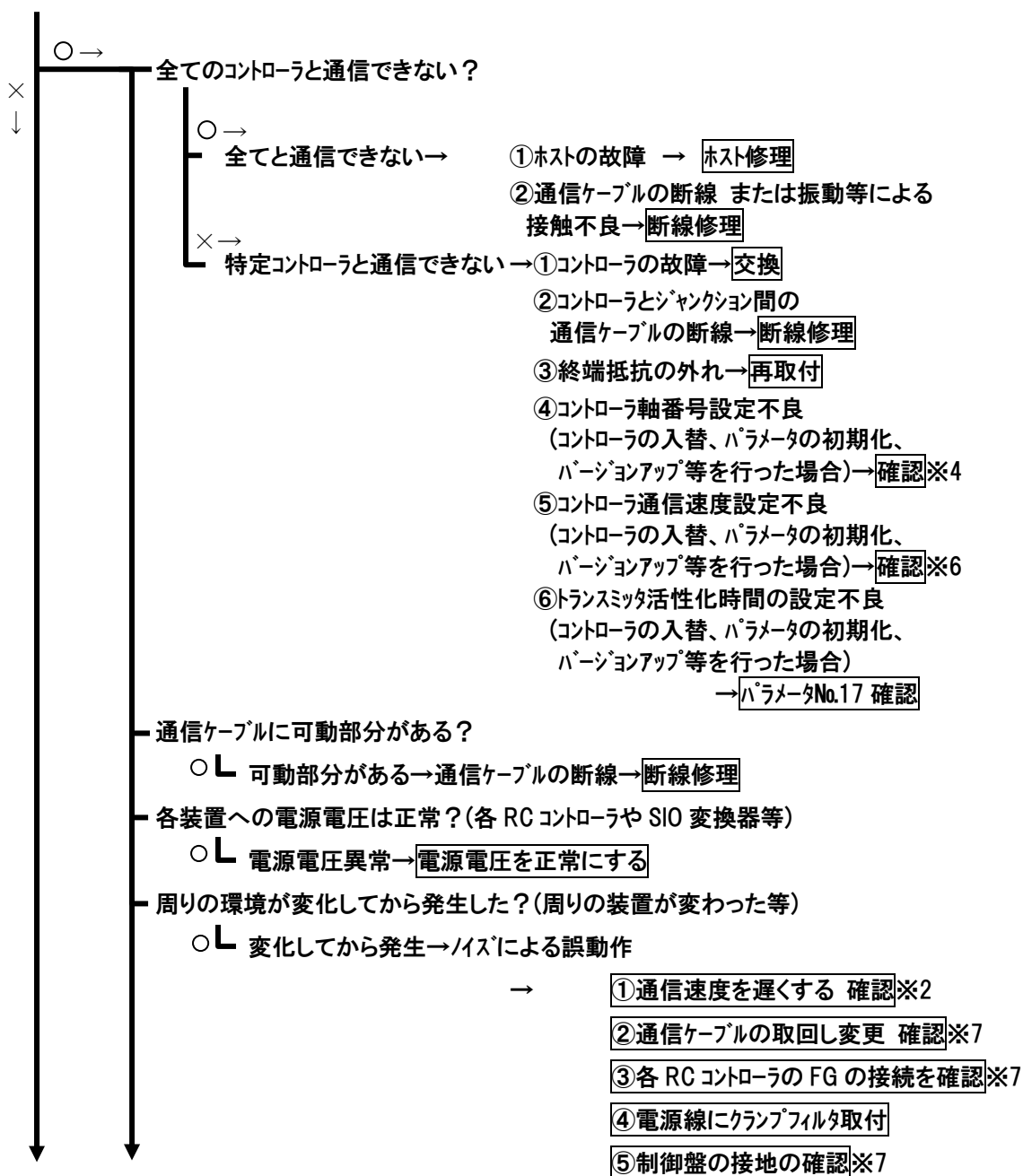
当てはまる項目を選択して □で囲まれた処理をおこなってください。

具体的な処理内容はフローチャートの後に記載していますので、※の示す内容をご確認ください。

○=YES、×=NO

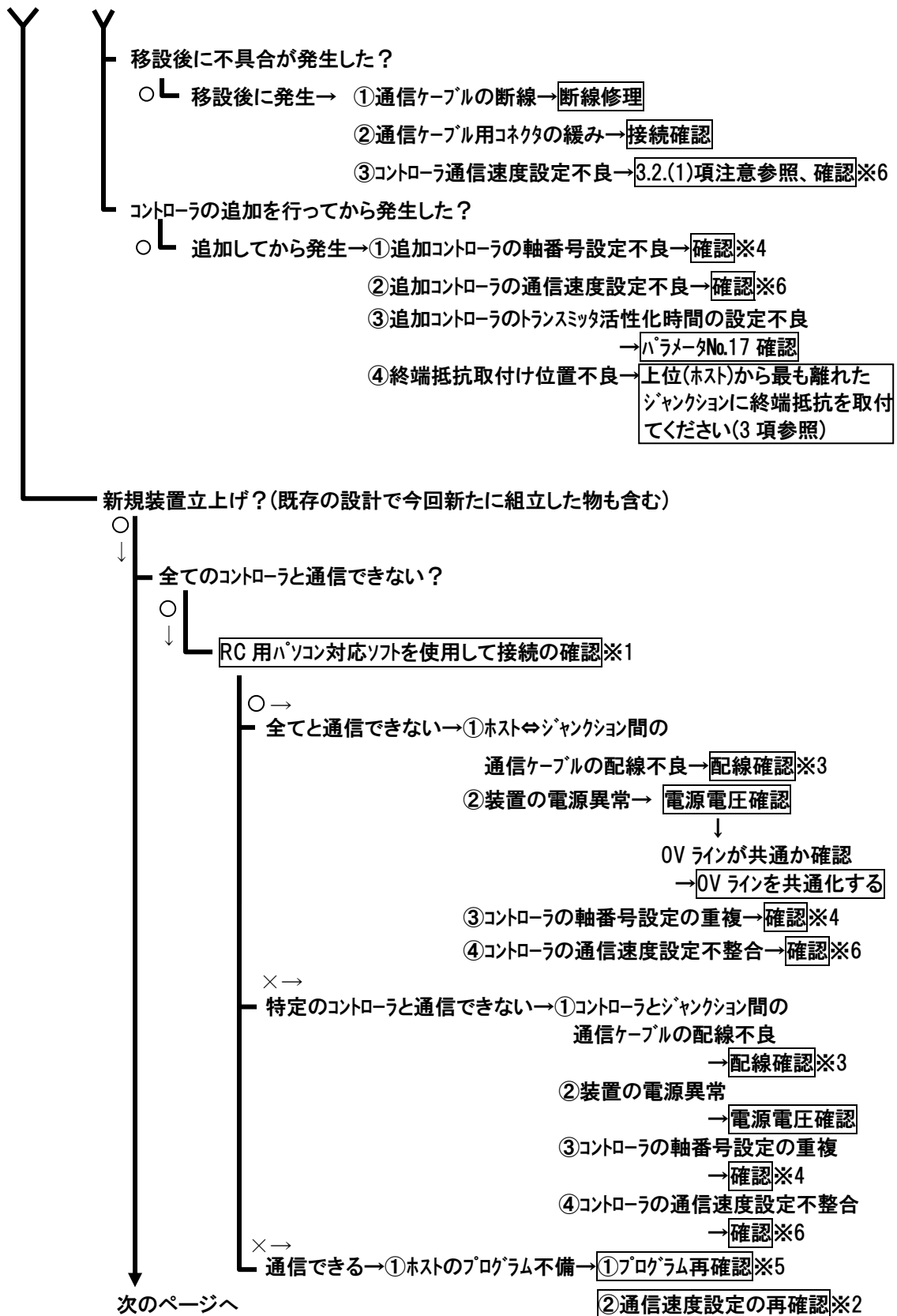
・症状：正常に通信できない！

今まで通信できていた？



次ページに続く

前ページより

トラブル
シューティング

前ページより

- Y
- 特定のコントローラと通信できない？
 - L ①コントローラとジャンクション間の通信ケーブルの配線不良→配線確認※3
 - ②装置の電源異常→電源電圧確認→0Vラインが共通になっているか確認
→0Vラインを共通化する
 - 時々通信できない？
 - L ①ノイズによる誤動作→
 - ①通信速度を遅くする 確認※2
 - ②通信ケーブルの取回し変更 確認※7
 - ③各 RC コントローラの FG の接続を確認※7
 - ④電源線にクランプフィルタ取付
 - ⑤制御盤の接地の確認※7
 - ②ホストのプログラム不備→プログラム再確認(通信バッファオーバーフロー等発生?)

※1 3.1 項、3.2 項、3.3 項を参考にホストにパソコンを接続してください。

- ① パソコン対応ソフトを起動してください。
- ② 設定→アプリケーション設定を選択します。

通信設定画面のポートが、使用しているパソコンのポート番号になっているか、最終軸 No.が接続されている軸の数以上に設定されているかを確認してください。

(設定が違っていた場合は、再設定を行い、RC 用パソコン対応ソフトを再起動してください。)

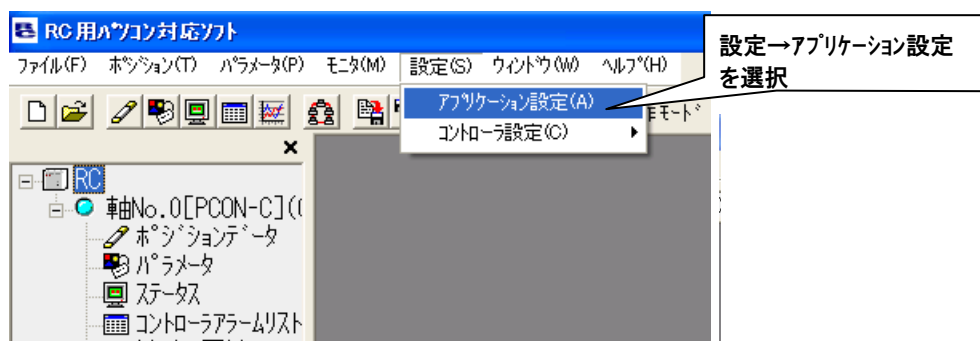


図 9.1

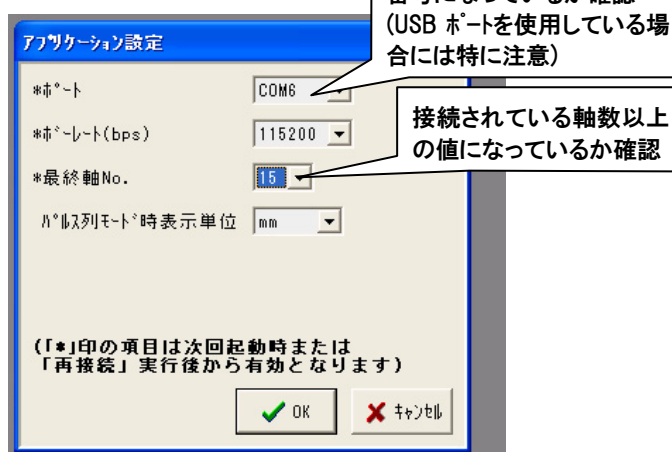


図 9.2

③ポジション→編集/ティーチを選択します。

ポジションデータ編集軸選択画面が表示され、その中に接続されている軸が表示されています。接続されている軸番号が表示されている軸は、通信ができています。

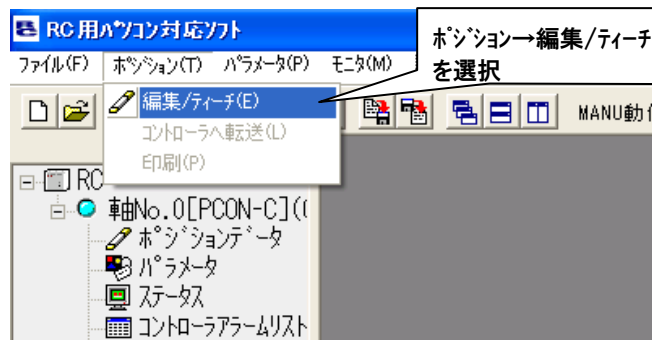


図 9.3

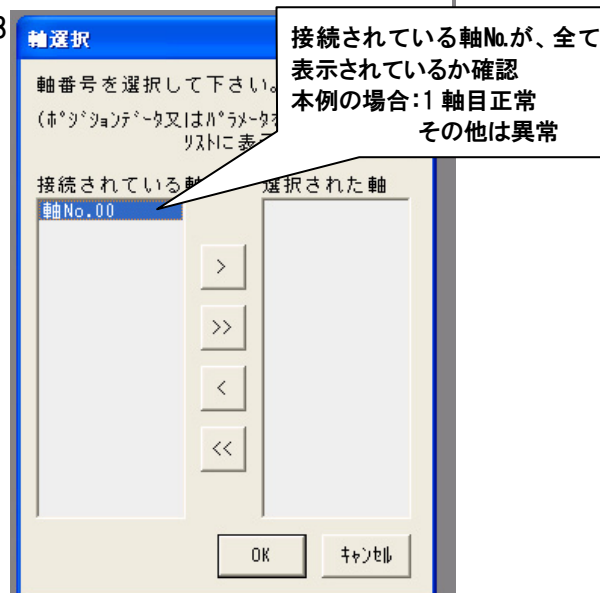


図 9.4

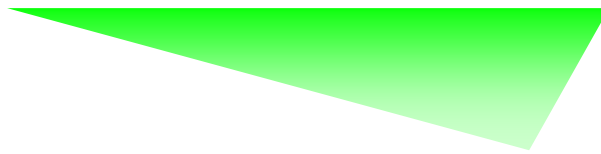
- ※2 3.6 項を参考に通信速度の設定を遅くなるように設定します。
- ※3 3.1 項、3.2 項、3.3 項を参考に配線の再確認をしてください。
- ※4 3.5 項を参考に軸番号の設定を再度確認してください。(重複しないように設定されているか確認)
- ※5 3.4 項の手順が守られているか再度確認してください。
 - ①ファンクションコード 03 のケリ以外を使用している場合は、5.4.16 項(RTU)、6.5.16 項(ASCII)のPIO/Modbus 切替を Modbus 側にしてあるか確認してください。
 - ②RC 用パソコン対応ソフトを使用して、RC コントローラの再起動を行わないと RC 用パソコン対応ソフトを接続した時の通信速度設定のままになっています。その場合には、RC コントローラを再起動してください。

- ※6 3.6 項を参考に通信速度の設定を再確認してください。
全ての RC コントローラ、ホストを同じ通信速度設定にしてください。
- ※5 の②を確認してください。
- ※7 動力線やパルス信号を送っているような配線と平行にならないように通信ケーブルを配線してください。
通信ケーブルのシールド処理は適切に行っているか確認してください。(推奨:1 点アース)
各 RC コントローラの取扱説明書にある設置環境 及びノイズ対策の内容が施されているかを確認してください。

以上のご確認で改善しない場合、当社までご相談下さい。

尚、この際には発生している現象 およびフローチャートでの確認事項を併せて、ご連絡くださいますようお願い致します。

8 参考資料



8.1 CRC チェック計算

C言語による CRC 値計算用のサンプル関数を示します。

公開されている Modbus プロトコル仕様書(PI-MBUS-300 Rev.J)に記載されている CRC 値計算用の関数と同等のものです。

```

unsigned short CalcCRC16swap(
    unsigned char*  puchMsg,                /* message to calculate */
    unsigned short usDataLen)              /* quantity of bytes in message */
{
    unsigned char  uchCRCHI = 0xFF;        /* high byte of CRC initialized */
    unsigned char  uchCRCLo = 0xFF;       /* low byte of CRC initialized */
    unsigned int   uIndex;                 /* will index into CRC lookup table */

    while(usDataLen-->0)                  /* pass through message buffer */
    {                                       /* calculate the CRC */
        uIndex = uchCRCHI ^ *puchMsg++;
        uchCRCHI = uchCRCLo ^ auchCRCHI[uIndex];
        uchCRCLo = auchCRCLo[uIndex];
    }
    return (uchCRCHI << 8 | uchCRCLo);
}

```

```

const unsigned char auchCRCHI[] =
/* Table of CRC values for high-order byte */
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,

```



```

0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
};

```

```

const unsigned char auchCRCLo[] =
{ /* Table of CRC values for low-order byte */
0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7, 0x05, 0xC5, 0xC4, 0x04,
0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8,
0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,
0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0xD6, 0xD2, 0x12, 0x13, 0xD3, 0x11, 0xD1, 0xD0, 0x10,
0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32, 0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4,
0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,
0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C,
0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26, 0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0,
0xA0, 0x60, 0x61, 0xA1, 0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,
0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68,
0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C,
0xB4, 0x74, 0x75, 0xB5, 0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54,
0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98,
0x88, 0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83, 0x41, 0x81, 0x80, 0x40,
};

```

8.2 SIO と PIO の併用システム構成

RC コントローラを PIO で駆動しておき、SIO(通信)で現在の位置等をモニタすることができます。モニタが可能なのは、RTU/ASCII 共にファンクションコード 03 を使用するクエリとなります。5.4.16 または 6.5.16 PIO/Modbus 切替を PIO 側に設定、及び モード SW がある RC コントローラに関しては AUTO 側に設定してご使用ください。

PIO と SIO を併用できる RC コントローラは、以下となります。

- ・PCON-C/CG/CF/CA/CFA、PCON-CY、PCON-PL/PO、
- ・ACON-C/CG、ACON-CY、ACON-PL/PO、
- ・SCON-C/CA、
- ・ERC2

SIO と PIO の併用システム構成例 1

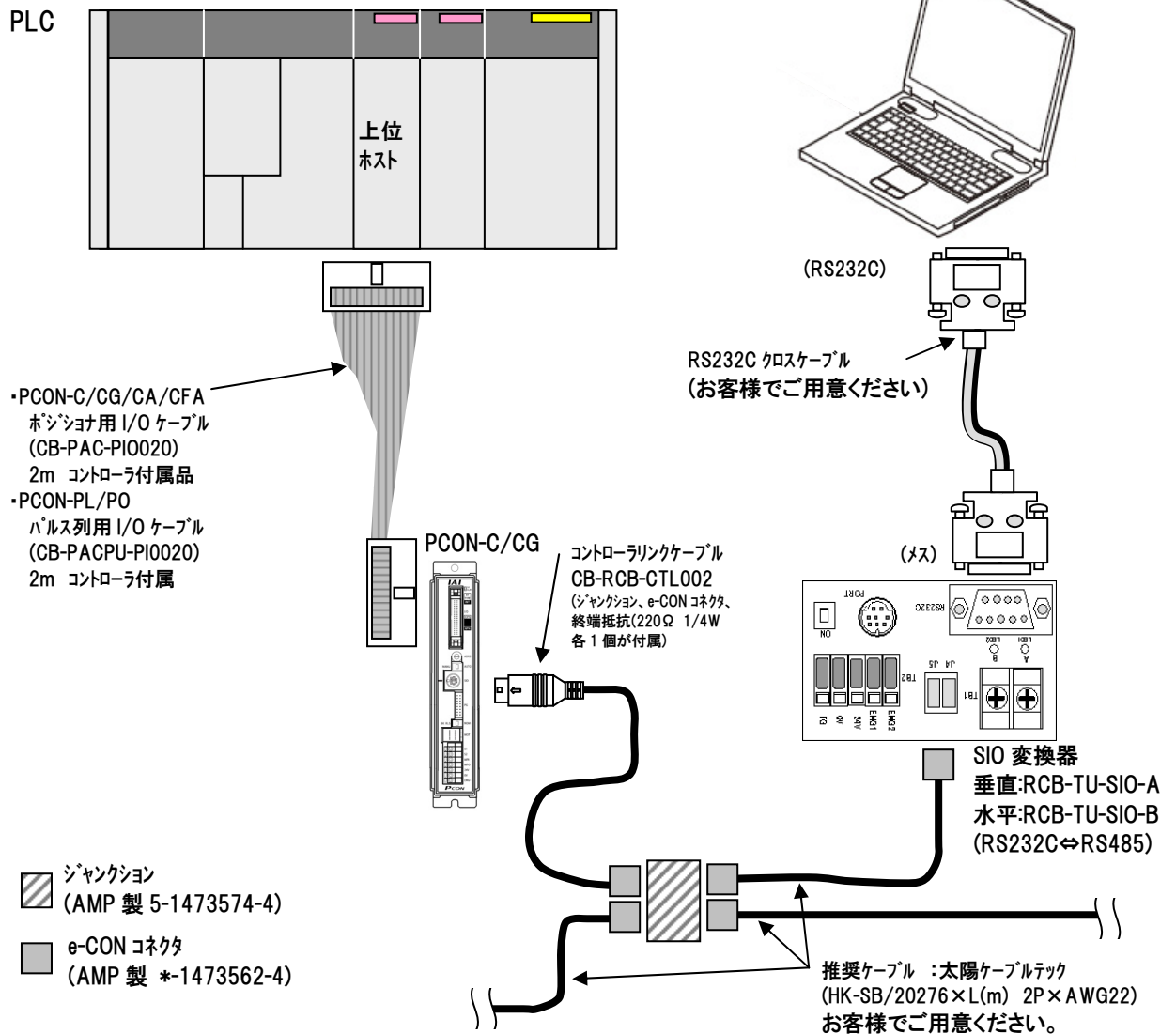
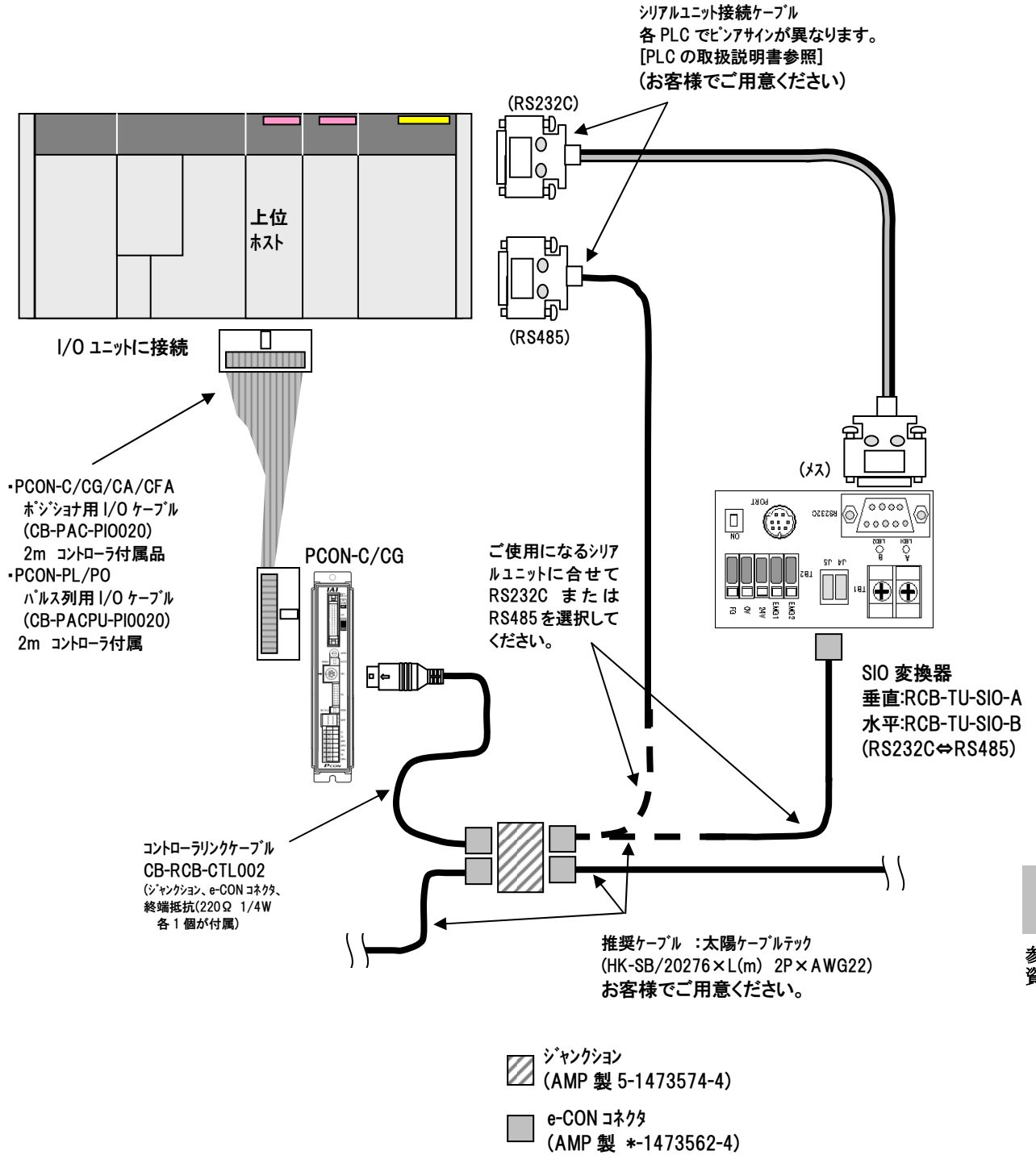


図 11.1

SIO と PIO の併用システム構成例 2



参考
資料

図 11. 2

変更履歴

改定日	改訂内容
2011.6	第4版リリース ・安全がた追加 ・06ケリの表記見直し ・10ケリの表記見直し
2011.10	第5版リリース ・対応機種に SCON-CA を追加 (ロードセルキャリブレーション指令、完了、計測値読み込みコマンド、レジスタの追加)
2012.10	第6版リリース ・対応機種に ERC3、PCON-CA/CFA を追加 (メンテナンス情報の読み込みコマンド、レジスタの追加)
2013.6	第7版リリース ・ポジションデータ読み出しコマンド追加、メッセージレベルエラー出力有無による置き換え時の注意を冒頭に記載

管理番号 :MJ-0162-7A(2013年6月)



株式会社 **アイエイアイ**

本社・工場	〒424-0103 静岡県静岡市清水区尾羽 577-1	TEL 054-364-5105	FAX 054-364-2589
東京営業所	〒105-0014 東京都港区芝 3-24-7 芝エクスージビルディング 4F	TEL 03-5419-1601	FAX 03-3455-5707
大阪営業所	〒530-0002 大阪市北区曽根崎新地 2-5-3 堂島 TSS ビル 4F	TEL 06-6457-1171	FAX 06-6457-1185
名古屋営業所	〒460-0008 名古屋市中区栄 5-28-12 名古屋若宮ビル 8F	TEL 052-269-2931	FAX 052-269-2933
盛岡営業所	〒020-0062 岩手県盛岡市長田町 6-7 クリエ 21 ビル 7F	TEL 019-623-9700	FAX 019-623-9701
仙台営業所	〒980-0802 宮城県仙台市青葉区二日町 14-15 アミ・グランデ二日町 4F	TEL 022-723-2031	FAX 022-723-2032
新潟営業所	〒940-0082 新潟県長岡市千歳 3-5-17 センザイビル 2F	TEL 0258-31-8320	FAX 0258-31-8321
宇都宮営業所	〒321-0953 栃木県宇都宮市東宿郷 5-1-16 ルーセントビル 3F A	TEL 028-614-3651	FAX 028-614-3653
熊谷営業所	〒360-0847 埼玉県熊谷市龍原南 1 丁目 312 番地あかりビル 5F	TEL 048-530-6555	FAX 048-530-6556
茨城営業所	〒300-1207 茨城県牛久市ひたち野東 5-3-2 ひたち野うしく池田ビル 2F	TEL 029-830-8312	FAX 029-830-8313
多摩営業所	〒190-0023 東京都立川市柴崎町 3-14-2BOSEN ビル 2F	TEL 042-522-9881	FAX 042-522-9882
厚木営業所	〒243-0014 神奈川県厚木市旭町 1-10-6 シャンロック石井ビル 3F	TEL 046-226-7131	FAX 046-226-7133
長野営業所	〒390-0852 長野県松本市島立 943 ハーモネートビル 401	TEL 0263-40-3710	FAX 0263-40-3715
甲府営業所	〒400-0031 山梨県甲府市丸の内 2-12-1 ミサトビル 3 F	TEL 055-230-2626	FAX 055-230-2636
静岡営業所	〒424-0103 静岡県静岡市清水区尾羽 577-1	TEL 054-364-6293	FAX 054-364-2589
浜松営業所	〒430-0936 静岡県浜松市中区大工町 125 大発地所ビルディング 7F	TEL 053-459-1780	FAX 053-458-1318
豊田営業所	〒446-0056 愛知県安城市三河安城町 1-9-2 第二東洋ビル 3F	TEL 0566-71-1888	FAX 0566-71-1877
金沢営業所	〒920-0024 石川県金沢市西念 3-1-32 西清ビル A 棟 2F	TEL 076-234-3116	FAX 076-234-3107
京都営業所	〒612-8401 京都市伏見区深草下川原町 22-11 市川ビル 3 F	TEL 075-646-0757	FAX 075-646-0758
兵庫営業所	〒673-0898 兵庫県明石市榑屋町 8 番 34 号大同生命明石ビル 8F	TEL 078-913-6333	FAX 078-913-6339
岡山営業所	〒700-0973 岡山市北区下中野 311-114 OMOTO-ROOT BLD. 101	TEL 086-805-2611	FAX 086-244-6767
広島営業所	〒730-0802 広島市中区本川町 2-1-9 日宝本川町ビル 5F	TEL 082-532-1750	FAX 082-532-1751
松山営業所	〒790-0905 愛媛県松山市榑味 4-9-22 フォーレスト 21 1F	TEL 089-986-8562	FAX 089-986-8563
福岡営業所	〒812-0013 福岡市博多区博多駅東 3-13-21 エフビル WING 7F	TEL 092-415-4466	FAX 092-415-4467
大分出張所	〒870-0823 大分県大分市東大道 1-11-1 タンネンバウム III 2F	TEL 097-543-7745	FAX 097-543-7746
熊本営業所	〒862-0954 熊本県熊本市神水 1-38-33 幸山ビル 1F	TEL 096-386-5210	FAX 096-386-5112

お問い合わせ先

アイエイアイお客様センター エイト

(受付時間) 月~金 24 時間 (月 7:00AM~金 翌朝 7:00AM) 土、日、祝日 8:00AM~5:00PM (年末年始を除く)	
フリー コール	0800-888-0088
FAX:	0800-888-0099 (通話料無料)

ホームページアドレス <http://www.iai-robot.co.jp>

IAI America Inc.

Head Office: 2690 W. 237th Street Torrance, CA 90505
 TEL (310) 891-6015 FAX (310) 891-0815
 Chicago Office: 1261 Hamilton Parkway Itasca, IL 60143
 TEL (630) 467-9900 FAX (630) 467-9912
 Atlanta Office: 1220 Kennestone Circle Suite 108 Marietta, GA 30066
 TEL (678) 354-9470 FAX (678) 354-9471
 website : www.intelligentactuator.com

IAI Industrieroboter GmbH

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany
 TEL 06196-88950 FAX 06196-889524

IAI (Shanghai) Co., Ltd.

SHANGHAI JIAHUA BUSINESS CENTER A8-303, 808, Hongqiao Rd. Shanghai 200030, China
 TEL 021-6448-4753 FAX 021-6448-3992
 website : www.iai-robot.com

製品改良のため、記載内容の一部を予告なしに変更することがあります。
 Copyright © 2011, Oct. IAI Corporation. All rights reserved.